Using Product Tolerances to Drive Manufacturing System Design

by

Daniel D. Frey

B.S., Aeronautical Engineering, Rensselaer Polytechnic Institute, 1987
M.S., Mechanical Engineering, University of Colorado, 1993

Submitted to the Department of Mechanical Engineering
in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy
at the
Massachusetts Institute of Technology
June 1997

© 1997 Daniel D. Frey. All rights reserved.

The author hereby grants to MIT permission to reproduce
and to distribute publicly paper and electronic
copies of this thesis document in whole or in part.

Signature of Author:

Department of Mechanical Engineering
May 27, 1997

Certified by:

Kevin N. Otto
Robert Noyce Career Development 'sistant Professor of Mechanical Engineering
Thesis Supervisor

Accepted by:

Ain A. Sonin
Professor of Mechanical Engineering
Chairman, Departmental Graduate Committee

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

JUL 2 1 1997

ARCHIVES

LIBRARIES

Using Product Tolerances to Drive Manufacturing System Design

by

Daniel D. Frey

Submitted to the Department of Mechanical Engineering
on May 9, 1997 in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

Abstract

This thesis develops analytical tools for use in the design of manufacturing systems to ensure compliance with product tolerances. The thesis introduces new concepts for producibility analysis -- a process capability matrix and bias vector. These are ordered sets of dimensionless parameters that capture information on a manufacturing system's response to noise factors. Algorithms and equations are derived that use the matrix to estimate the yield of a manufacturing system. The methods prove to be accurate on engineering problems for which all other known techniques are inadequate because they do not account for statistical correlation among product acceptance criteria. The process capability matrix also proves useful in block diagram representations of production systems. The block diagrams are shown to be useful in evaluating the effectiveness of on-line adjustment strategies for variation reduction. Three industrial case studies are employed to evaluate the effectiveness of the representation and associated algorithms as tools for design decision making. The case studies concern surface mount of multichip modules, dual head valve grinding, and CNC crankpin grinding.

Thesis Advisor:	Kevin N. Otto
Committee Members:	Seth Lloyd
	Sanjay E. Sarma
	Warren P. Seering
	Alexander H. Slocum

Acknowledgments

I am grateful to my advisor, Kevin Otto, for his help and encouragement. He always gave me the freedom to pursue my interests.

I owe thanks to each member of my thesis committee. Alex Slocum was essential to my intellectual development. He deserves a large measure of credit for defining the problem and getting the project rolling. Warren Seering helped to place the work a wider perspective. Seth Lloyd was a valuable resource on information theory and control. Sanjay Sarma helped me to understand the importance of fixturing in manufacture. Other MIT faculty members who have influenced my thinking about design and manufacture include David Wallace, Woodie Flowers, Anna Thornton, and Nam Suh.

Many MIT staff members were a tremendous help to me in administration, laboratory, and library support. I'd especially like to thank Dick Fenner, Leslie Regan, Susan Barnaby Travis, Suzanne Weiner, Bob Nutall, Norm MacAskill, Gerry Wentworth, Kate Melvin, Maggie Lynch, Maureen Lynch, and Carol Robinson.

Industrial collaborators have had a substantial influence on this work. I want to thank Joe Wysocki, Leslie Momoda, Stan Taketani, Kelly Asada, and Chris DeGroot of Hughes Electronics for their help in bringing together the MCM surface mount case study. I also owe thanks to Bill Pflager and Tim Hykes at Landis and Hans Soons and Don Blomquist at the National Institute for standards and Technology for help with the machining case studies.

The students at MIT create an exciting intellectual environment from which I've profited tremendously. Ross Levinsky taught me to not drag my feet. Joost Bonson got me excited about enterpreneurship. Bill Singhose helped me to understand feedforward control. My lab mates Raj Suri, Marc Zemel, Brian Welker, Chris Paynter, and Javier Gonzalez were great sounding boards for ideas.

My family kept me sane throughout this process. My wife, Vicki, put up with months of my being away at factories. Mom, thanks for letting me be the P&J. Phil, wasn't this your idea? Randy, thanks for helping me to understand that the technical part isn't everything. Steve, Andy, Amy, Ben, David, Jon, Grandpa, Gramma, thanks for everything.

Biographical Note

Dan Frey was born in 1966 in Buffalo New York. From 1983 to 1987, he studied Aeronautical Engineering at Rensselaer Polytechnic Institute on a Naval ROTC scholarship. Upon graduation, he received the Secretary of the Navy Distinguished Graduate Award as the top Naval ROTC graduate of RPI.

In 1987, Dan was commissioned as an Ensign and began primary flight training in the T-34C Turbine Mentor. He was selected for jet training and went on to carrier qualify aboard the USS Lexington in June 1989 in a T-2C Buckeye. He also flew the T-A4J Skyhawk. In 1990, Dan was assigned to the U.S. Southern Command in Panama where he provided intelligence support of counternarcotics operations. He received a Joint Service Commendation Medal for his achievements in this role. Dan left the Navy as a Lieutenant in 1991.

From 1992-1993, Dan earned a Master's degree in Mechanical Engineering at the University of Colorado in Boulder. With his thesis advisor, Lawrence Carlson, he developed three new prosthetic prehensors for upper limb amputees. During this period, Dan served as a design consultant to Therapeutic Recreation Systems, Inc. Some of his designs are still in production as of the date of this publication. Dan also earned a license as an automotive emissions mechanic from the State of Colorado. Dan's work at CU led to the following journal publications and patents:

Frey, D. D., L. E. Carlson, V. Ramaswamy, and B. Radocy, 1995, "Voluntary Opening Prehensors with Adjustable Grip Force," *Journal of Prosthetics and Orthotics*, vol. 7, no. 4, pp. 124-131.

Carlson, L. E., B. Veatch, and D. D. Frey, 1995, "Efficiency of Prosthetic Cabl and Housing," *Journal of Prosthetics and Orthotics*, vol. 7, no. 3, pp. 96-99 .

Frey, D. D. and L. E. Carlson, 1994, "A Body Powered Prehensor with Variable Mechanical Advantage," *Prosthetics and Orthotics International*, vol. 18, pp. 118-123.

Carlson, L. E., D. D. Frey, and E. S. Brown, "Locking Mechanism for a Voluntary Closing Prosthetic Prehensor," application # 08-804-545 pending.

From 1994-1997, Dan studied mechanical engineering at MIT. In collaboration with his advisor and thesis committee, he has developed analytical techniques for design of manufacturing systems (as documented in this dissertation). During his stay at MIT, Dan was head TA for 2.70, a guest researcher at the National Institute for Standards and Technology, a consultant to Landis grinding machines, and a Hughes Doctoral Fellow. Dan's work in manufacturing has led to the following journal publications and patents:

Frey, D. D., K. N. Otto, and W. Pflager (sponsored by R. Donaldson), 1997, "Swept Envelopes of Cutting Tools in Integrated Machine and Workpiece Error Budgeting," to appear in *Annals of the CIRP*.

Frey, D. D., and T. Hykes, "A Method for Virtual Machining," application # 08-581-37 granted, to issue.

Dan and Victoria Frey have been happily married for six years. Vicki has a B.S. in Ecology and Environmental and Population Biology from the University of Colorado. Vicki is currently the Data Manager for the Endangered Species Program at the Massachusetts State Division of Fisheries and Wildlife.

Dan enjoys tennis, skiing, and running. He is a private pilot with over 500 hours flight time. Dan is also an avid student of philosophy, especially philosophy of science.

Table of Contents

List of Figures

List of Tables

1. Introduction

This introductory chapter defines the goal of the thesis, discusses the nature of the problem, and sketches my approach to its solution. In sketching the problem this chapter alludes briefly to each of the three major case studies.

1.1 The Goal

In the book, *The Goal*, Eliyahu Goldratt reminded manufacturing engineers and managers that we should always use our top level goals to drive our activities. This applies to writing a thesis no less than to running a manufacturing system. Therefore, this thesis begins by succinctly stating its goal.

The goal of this thesis is to help engineers design manufacturing systems whose products conform to their dimensional tolerances.

It is worthwhile to elucidate this goal further. When I say that I wish to **help engineers design**, I mean that I want to provide analytical engineering tools. Analytical tools may be used for selection among proposed alternatives, to guide evolution of design alternatives, and may even allow some synthesis problems to be recast as analysis problems. However, it is *not* my goal to provide prescriptions for the design process. This thesis does not seek to advance a design methodology.

Manufacturing systems may be defined as an arrangement of processes that transform materials, energy, and information into products. The processes that comprise a manufacturing system may include forming, casting, machining, assembly, etc. However, I wish to use the term more broadly to include the process of accepting or rejecting incoming material, the process of specifying and maintaining the performance of production equipment, the process of measuring and removing variation and bias, and the process of inspecting and testing the product. All of these have a significant effect on the quality of the final product and on the profitability of the

manufacturing enterprise. Therefore, I submit that these processes should be designed concurrently and not appended to an existing manufacturing system design.

Dimensional tolerances are the engineering specifications that determine acceptability of the product. I use the term "dimensional tolerances" as opposed to "specifications" because the case studies in the thesis all focus on product geometry as opposed to quality characteristics such as fracture strength or electrical conductivity. However, the goal of this thesis is to permit consideration of any measurable product characteristics used in the product's acceptance criteria.

Another way of refining the goal stated above is to consider the types of questions that engineers need to answer in designing manufacturing systems. Some questions that this thesis will enable engineers to answer include:

- Of the proposed system designs, which one will achieve the lowest defect rates?
- Which sources of noise are the greatest contributors to the quality problems in this system?
- What is the smallest set of quality characteristics that I need to inspect to reduce the risk of shipping defective products to an acceptable level?
- What set of variables can I use to adjust a set of quality characteristics back onto target?
- What variables should I measure and use to compute on-line adjustments?
- How accurately do I have to measure the variables and how precisely must I perform the adjustments for the scheme to be effective?
- What pattern of error will exist across the surface a workpiece machined under known conditions?
- What is the most likely cause of an observed pattern of error on a machined workpiece?

1.2 The Nature of Manufacturing System Design

A good start to solving any problem is to understand its nature. What are the essential features of the problem? What are the critical issues to be addressed? What special structure of the problem affords avenues for solution? To understand quality issues in manufacturing systems, one must grasp the nature of tolerances, variation, and system architecture. This section explores these three major areas and sneaks in a preview of each major case study in this thesis.

1.2.1 Product Tolerances, Multiple Outputs, and Dominance

Product tolerances are essential to manufacture. In fact, the development of toleranced engineering drawings was a key technology enabling the shift from hand craftsmanship to mass production [Jaikumar, 1988]. This section provides a brief review of concepts of tolerance in engineering and manufacture. It is based upon ANSI Y14.5M which sets forth the standards for engineering dimensioning and tolerancing.

In manufacture, there is always some variation in the dimensions of products. In practice, engineers manage this variation by defining tolerances on dimensions of features. A feature is "the general term applied to a physical portion of a part, such as a surface, hole, or slot." A dimension is "a numerical value expressed in appropriate units of measure ... to define the size or geometric characteristic, or both, of a part or feature." A tolerance is characterized by three values, a basic dimension, a low limit, and a high limit. The basic dimension is "a numerical value used to define the theoretically exact size, profile, orientation, or location of a feature or datum target. It is the basis from which permissible variations are established by tolerances on other dimensions..." The low limit is the "value of a dimension below which the part will be considered as a defect." Similarly, a high limit is the "value of a dimension above which the part will be considered as a defect."

As an example, consider the dimensioned drawing of an engine poppet valve in Figure 1.2.1. The basic dimension of the valve seat is $45^{\circ}31'$. The tolerances are specified as $\pm 20'$. Such *plus and minus tolerancing* implies that the basic dimension is in the center of the tolerance range. We will revisit the problem of satisfying the tolerance on this valve seat in Chapter 10.

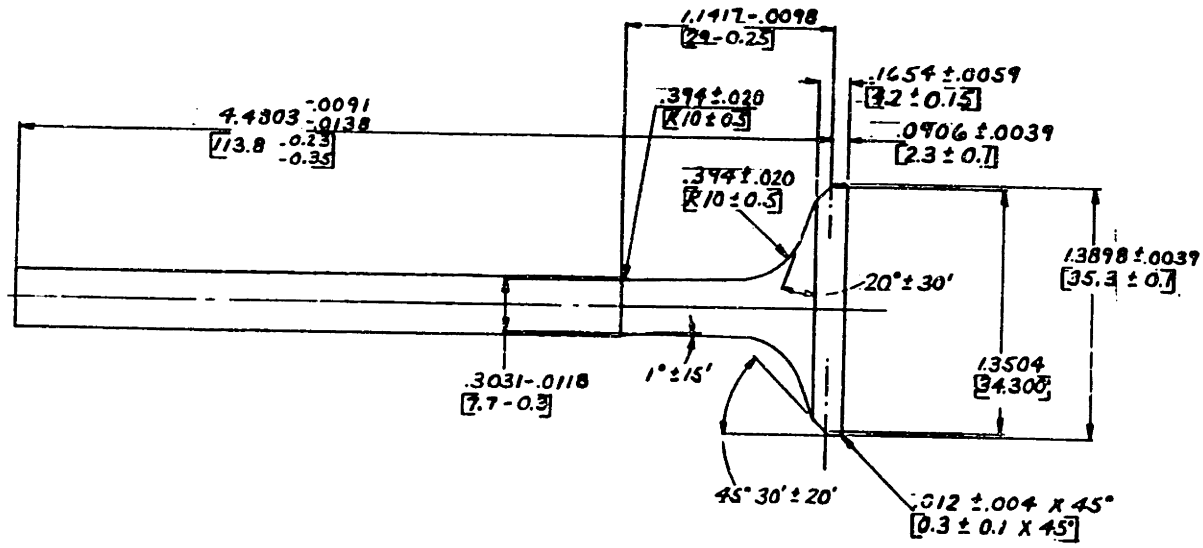


Figure 2.2.1 Dimensioned drawing of an engine poppet valve.

Tolerances of form are a special type of tolerance that require some additional definitions. A form tolerance defines a zone in which all the points on the surface of a part must lie. Form tolerances include straightness, flatness, circularity, cylindricity, and profile. Form tolerance are applicable to single (individual) features or elements of single features; therefore, form tolerances are not related to datums (ANSI uses "datums" as the plural of datum even though "data" is more correct). Since all tolerances are defined as bounds on a dimension, it logically follows that a "dimension of form" is the value of the tightest tolerance of form which a given part would meet. Techniques for managing tolerances of form are developed in Chapter 11 in the context of grinding automotive crankpins.

Now that we have reviewed tolerancing as defined by ANSI standards, let us explore the consequences of these definitions for manufacturing system design. There are three critical issues to address – the binary character of product acceptance, the multi-output nature of dimensional variation, and dominance.

As defined by ANSI, product acceptance is binary; Either a part is accurate enough or it is defective. This binary quality of acceptance is inherent to the manufacturing enterprise as it is conducted today. At some point, the part must either be accepted and assembled into the product or it must be considered defective and scrapped or reworked. This "goal post" mentality or "fraction defective fallacy" has been criticized [Phadke, 1989], and quality loss functions have

been proposed as an alternative (see Chapter 1.3.4). However, the binary nature of acceptance is still the reality in engineering practice today. It is often rational for the designers of manufacturing systems to take yield or fraction defective as the primary performance metric to optimize.

Products most often have multiple dimensions or quality characteristics that enter into their acceptance criteria. The relatively simple drawing in Figure 1.2.1 has over a dozen toleranced dimensions -- many engineering drawings have hundreds or thousands of toleranced dimensions. Thus, even manufacturing systems that make only one product can be considered multi-output systems.

Whenever a product has multiple toleranced dimensions (which is true in most cases) there exists a relationship of dominance among the dimensions. Any product which has a dimension outside the tolerance bounds is classified as a defect regardless of how many other dimensions of the same part conform to their tolerances or how closely they conform. The very definition of tolerances according to the ANSI standards incorporates this concept of dominance. There are generally very good engineering reasons for defining product acceptance in this way. No matter how accurate the valve in Figure 1.2.1 may be along other dimensions, if its valve seat angle is too far off, there will be a poor seal with the engine block.

The binary nature of product acceptance, the multi-output character of manufacture, and the dominance among these multiple binary criteria has significant implications for the designer of manufacturing systems. These considerations have a strongly affected my approach to manufacturing systems analysis.

1.2.2 Multiple Inputs, Correlation, and Mutual Information

The previous section discussed the means by which engineers define their tolerance of dimensional variation. This section addresses the sources of variation and the conditions that give rise to correlation among product dimensions.

Variation in product dimensions can be observed through repeated measurement. For example, a grinding machine of the type shown in Figure 1.2.2.1 was used to finish grind the crankpins on 50 crankshafts. The variation in diameter of the crankpins is in the histogram in

Figure 1.2.2.2. No matter how carefully one constructs or operates the grinding machine, with sufficiently accurate measurements, one may always reveal variation in the crankpins.

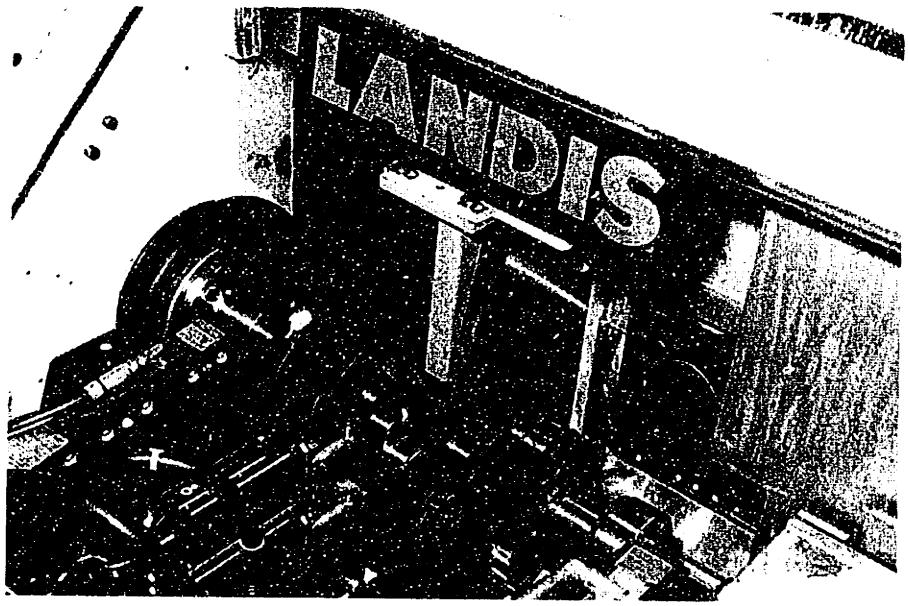


Figure 1.2.2.1 A CNC crankpin grinding machine.

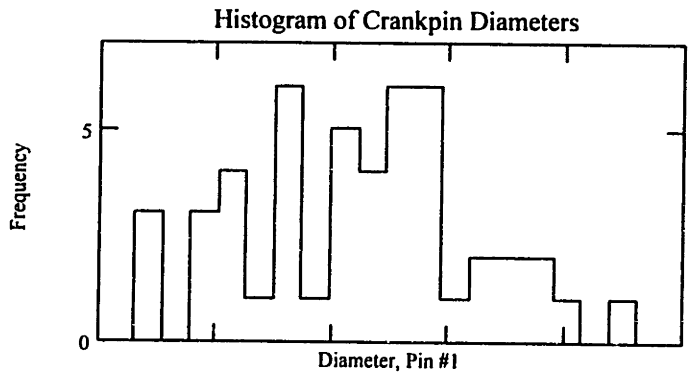


Figure 1.2.2.2 Variation in crankpin diameter.

There are usually multiple causes of the dimensional variation observed in products. Again consider the crankpin grinding process. Some of the causes of variation in the crankpins include:

- Grinding wheel attrition -- The grinding process requires, by its nature, that the surface of the grinding wheel be consumed. Naturally, this attrition will give rise to gradual rise in the size of the crankpins unless compensated by some means.

• Fixturing -- The crankshaft is supported by its main bearings. Any dimensional inaccuracies in the main bearings, therefore, will contribute to errors in the crankpins.

• Machine component errors – Many components of machine tools have dimensional inaccuracies that adversely affect product quality. For example, the lead screw that drives the wheelhead axis displays a once-per-revolution error that affects crankpin roundness.

The causes of dimensional variation in a manufacturing process frequently affect more than one dimension of a product. For example, in CNC crankpin grinding, wheel attrition will cause a size error in not just one crankpin, but probably all the other crankpins on the same crankshaft. Also, given this machine's configuration, the wheel attrition will also contribute to problems with the throw of the crankpins (distance of the center of the crankpin to the center of the main bearings).

The fact that individual causal factors affect multiple dimensions gives rise to correlation among dimensions. For example, the correlation between the size errors of neighboring crankpins is clearly evident in Figure 1.2.2.3.

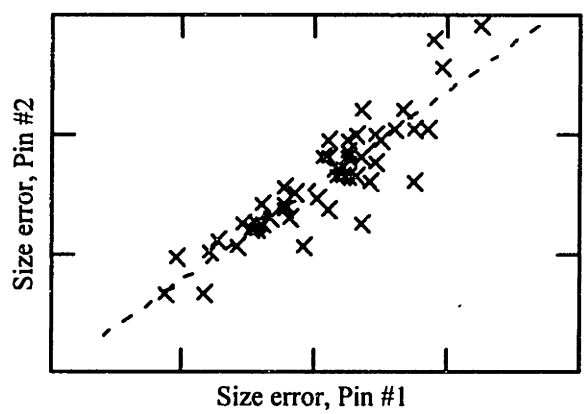


Figure 1.2.2.3 Correlation between the size errors of two crankpins.

Correlation among dimensions of products is not limited to features made by machining. Certainly, one should expect strong statistical correlation among multiple dimensions of a part made by a net shape process. For example, if variation in an injection molded part is due entirely

to uniform shrinkage, then the variation in all the part's linear dimensions will have correlation coefficients near unity.

One may argue that the only reason for the correlation observed in Figures 1.2.2.3 is that the part has multiple dimensions determined by a single process. You might argue that dimensions of a part made by two distinct processes should be uncorrelated. For example, if a part is machined on both a mill and a lathe, the variation in features generated by the mill ought to be statistically independent of those generated by the lathe. After all, the tool wear, thermal state, and squareness errors of the mill are not physically coupled to those of the lathe. However, if the features created by the lathe are used in *fixturing* the part for milling, then there may be significant statistical correlation between these features.

Correlation among quality characteristics seems to be the rule rather the exception. But what is the relevance to manufacturing system design? This thesis will show that correlation has significant impacts on defect rates, inspection design, and fault diagnosis.

Correlation combined with dominance can have a strong affect on defect rates. For example, consider a product with ten dimensions. Further, assume that the manufacturing system has a 99% chance of meeting each tolerance. Now consider the two limiting cases – complete statistical independence and perfect correlation among the ten dimensions. If each dimension is statistically independent of every other dimension, then the defect rate for that manufacturing system will be $1-0.99^{10}$ which is roughly 9.6%. If, however, the ten dimensions are perfectly correlated, then the defect rate will be 1% since acceptance on any one dimension ensure acceptance on all the others. So, for parts with a realistic number of specifications, failure to consider statistical correlation can result in gross miscalculation of the defect rate.

Correlation can also be an important consideration for designing inspection procedures. For example, let us imagine that we wish to be 99.9% certain of installing only acceptable crankshafts in our engines. We may employ inspection procedures to reach this level of certainty, but since inspection can be expensive, it behooves us to design an efficient inspection procedure. It is most likely wasteful to measure the size of every crankpin. The correlation among crankpin sizes implies that measuring the size of one crankpin provides information concerning the size of the other crankpins. An understanding of mutual information may allow reduced inspection costs while limiting risk.

Correlation can be the key to diagnosing the cause of degraded system performance. Correlation is, after all, an indication of a pattern in data. Just as doctors look for patterns of symptoms and match them to disease pathologies, machine trouble-shooters may employ correlation among part quality characteristics from post process gauging data to infer the cause of machine performance problems.

To recapitulate, statistical correlation among product dimensions is the rule rather than the exception. It should not be ignored. Rather, it should be captured in models and exploited to improve the design of the system.

1.2.3 Multiple Processing Steps, Adjustments, and System Architecture

Most products are manufactured by processes comprised of many steps each of which can impact final product quality. The variations introduced in one step may be added to, amplified, or reduced by subsequent processing steps. These interactions can have a significant impact on the performance of manufacturing systems as measured by yield, variance, and bias. Quality is a systems problem and manufacturing system designers should think carefully about system architecture. This section will explore these issues.

Consider the process of assembling electronic components onto a printed wiring board. Figure 1.2.3 depicts one of the final steps in the process, the placement of a component onto a board. Several manufacturing processes made this assembly step possible. The component body was fired. The ribbon leads around the circumference of the board were formed. The pads on the board were printed.

Each of these processing steps can have an effect on the results of the final assembly step. The goal is to position the ribbon leads to sit properly on the pads. If the component body is warped, it may be grasped incorrectly by the assembly robot. If the leads are malformed, they may not align with the pads. If the pads are too wide, there may be electrical contact with the neighboring lead.

Processing steps most often amplify or add to variation from earlier processing steps. However, in some cases, a processing step can reduce existing variation through an on-line adjustment. For example, the assembly robot in Figure 1.2.3 has a machine vision system that determines the locations of the individual ribbon leads. This measurement is used to optimally

reposition the component on the printed wiring board. This on-line adjustment may reduce the consequences of variations introduced upstream in the lead forming process.

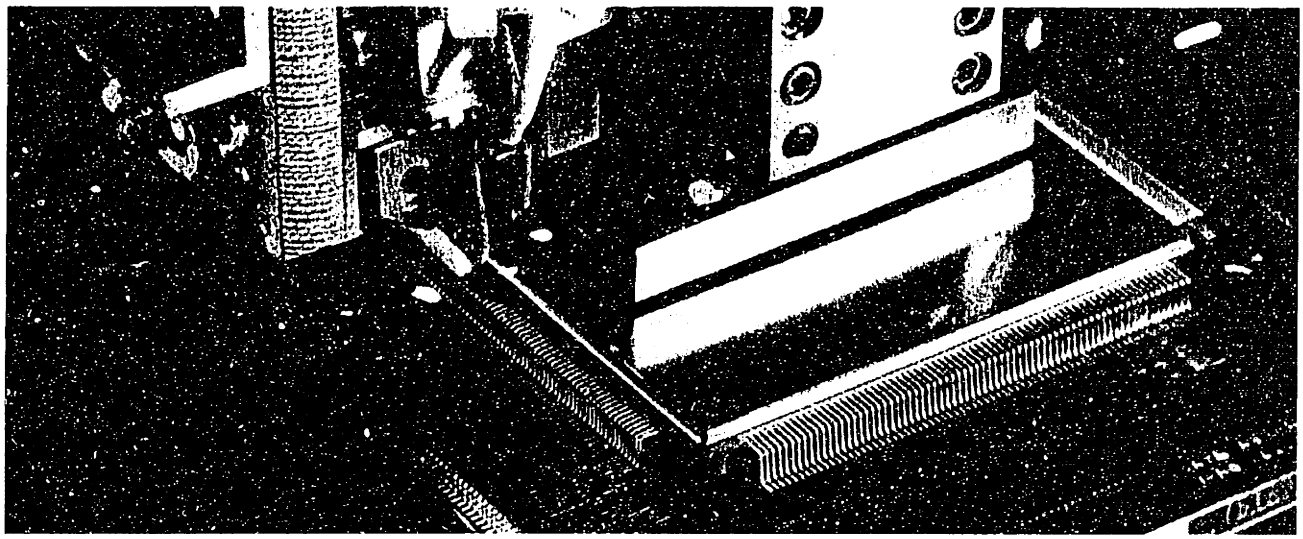


Figure 1.2.3 A robot assembles an electronic component onto a printed wiring board.

On-line adjustments such as the one described above are common in manufacture. Many airframe assembly processes involve shimming, all automobiles include adjustments for front end alignment, and many consumer optics products include adjustment screws used in post-assembly calibration.

Many design for manufacture texts suggest elimination of adjustments in manufacturing processes. Adjustments can lengthen cycle times and may add to quality problems if performed incorrectly [Anderson, 1990]. Overly general rules such as "eliminate adjustments" can be misleading. Adjustments are used in industry because, despite their drawbacks, they are often the most economical means to achieving required dimensional accuracy. Rather than blindly following rules, I submit that manufacturing engineers should employ analytical techniques to assess the effectiveness of adjustment procedures and decide on that basis whether to keep, modify, or eliminate an adjustment. In this way, engineers can select a small number of adjustments that substantially improve product quality at minimal cost.

1.3 The State of the Art

There exists a significant body of literature on issues of quality and dimensional variation in manufacture. Manufacturing enterprises use this body of knowledge and have managed to create good products for quite some time now. Most of them have done so without any help from me. So what the purpose of another dissertation on quality issues in manufacture? This is the question this section is intended to address. This section briefly reviews some of the major tools used to analyze manufacturing variation and points out areas in which they can benefit from improvement.

1.3.1 Process Capability Indices

The products of any manufacturing processes always exhibit *some* dimensional variation and bias. The question is not the existence of these phenomenon, but the capability of a process or system to manufacture acceptable products despite them. Therefore, engineers require measures of this capability to help them select processes, equipment, and materials.

Several indices are commonly employed to characterize the capability of a production system. Among the most common is the *process capability index* (C_p) -- a dimensionless ratio of the amount of variation that can be tolerated and the amount of variation present. It is defined as

$$C_p \equiv \frac{(U-L)/2}{3\sigma} \tag{1.3.1.1}$$

where U and L are upper and lower specification limits on a random variable and σ is the standard deviation of the random variable. Typically, the random variable represents a toleranced dimension or other quality characteristic that is used in a product's acceptance criteria. Figure 1.3.1.1 illustrates some of the terms in the definition of the process capability index. The random variable that represents the quality characteristic is denoted as q. The variation in q is described by a probability density function $p(q)$.

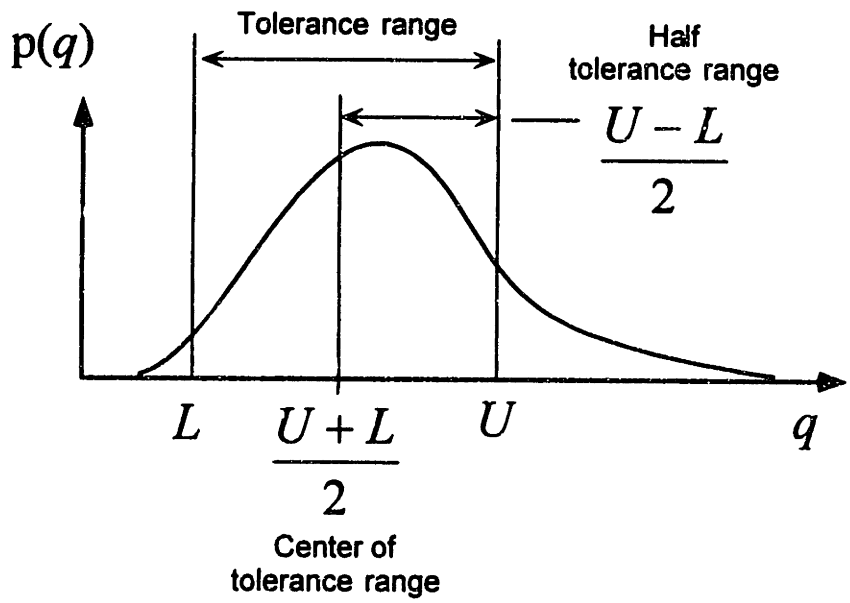


Figure 1.3.1.1. Process capability and yield.

Variation is not the only factor that tends to degrade the performance of a production system. *Bias*, the difference between the mean value of the quality characteristic and the center of the tolerance range, also tends to reduce the yield. The process capability index can be modified to include the effect of a mean shift. This leads to the definition of a *performance index* (C_{pk}) [Kane, 1986].

$$C_{pk} \equiv C_p(1-k). \tag{1.3.1.2}$$

where k is a dimensionless ratio of the absolute value of the bias and tolerance width

$$k \equiv \frac{\left|\mu - \frac{U+L}{2}\right|}{(U-L)/2}. \tag{1.3.1.3}$$

The performance index (C_{pk}) gives an indication of the expected performance of a manufacturing system under the influence of both variation and bias. The process capability index (C_p) gives an indication of the potential performance of a process if bias were entirely removed but variation were still present.

As an example, let us apply the concept of process capability indices to the surface mount process. Figure 1.3.1.2 displays a histogram of data collected from the electronics manufacturing process briefly discussed in Section 1.2. The measurements displayed are errors in the positions of individual ribbon leads. The vertical lines indicate the tolerances applied to the data.

Computing the standard deviation of this population and applying Equation 1.3.1.1, we find this process has a process capability $C_p = 0.82$. This is generally considered a low process capability (low capability is bad!); Many process engineers set a goal that process capability should meet or exceed 1.3. The low process capability can be observed directly in Figure 1.3.1.2; the spread in the data is fairly large compared to the tolerance width.

Computing the mean of the population and applying Equation 1.3.1.3, we find this process has a bias factor $k = 0.08$. This is considered a low value of bias (low bias is good!) given the low process capability; Some texts on quality in manufacture consider a 1.5 sigma mean shift to be typical [Juran, 1951] so that one would expect the bias factor to be the inverse of two times the process capability. Combining the process capability and bias factor into Equation 1.3.1.2 gives a performance index $C_{pk} = 0.75$. This is considered a rather poor performance index; Most manufacturing engineers shoot for performance indices greater than one.

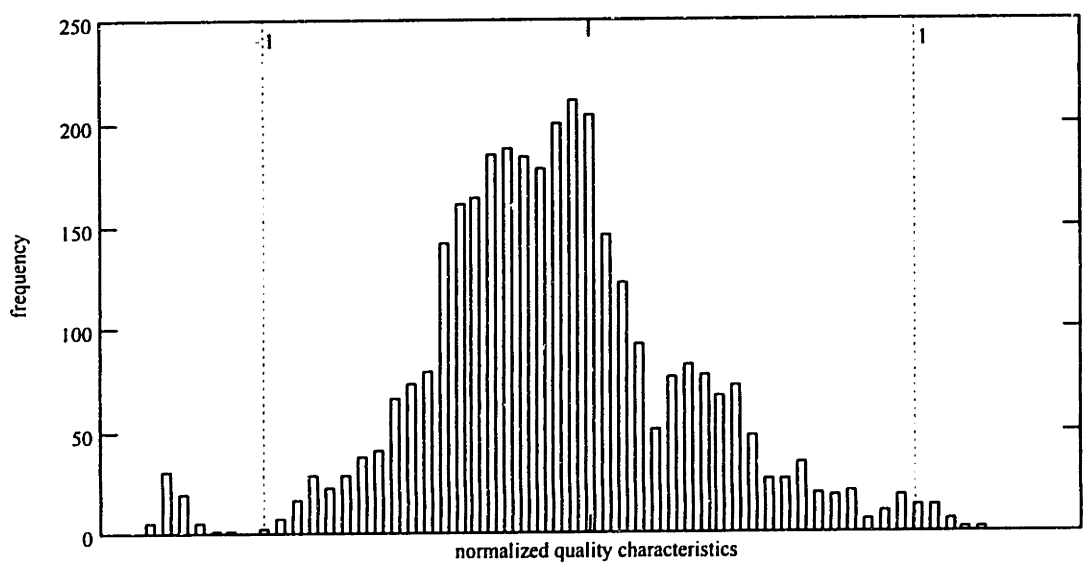


Figure 1.3.1.2 A histogram of data from an electronics manufacturing system.

1.3.2 First Time Yield

As discussed in section 1.2.1, most products have multiple quality characteristics to which tolerances are applied. This implies that there are multiple *opportunities* for a product to be

considered defective. The probability that any given quality characteristic meets its tolerances is sometimes called *first time yield* (Y_{FT})

$$Y_{FT} \equiv \Pr(L \le q \le U) \tag{1.3.2.1}$$

where q denotes the quality characteristic of interest and U and L are its upper and lower tolerance limits.

The quality characteristic q may be modeled as a random variable. That is, it may be considered as a variable that takes on different values in successive statistical trails conducted under apparently equal conditions. Every continuous random variable can be characterized by a probability density function $p(q)$ which, by definition, has the property that

$$Y_{FT} = \int_L^U p(q)\mathrm{d}q \ . \tag{1.3.2.2}$$

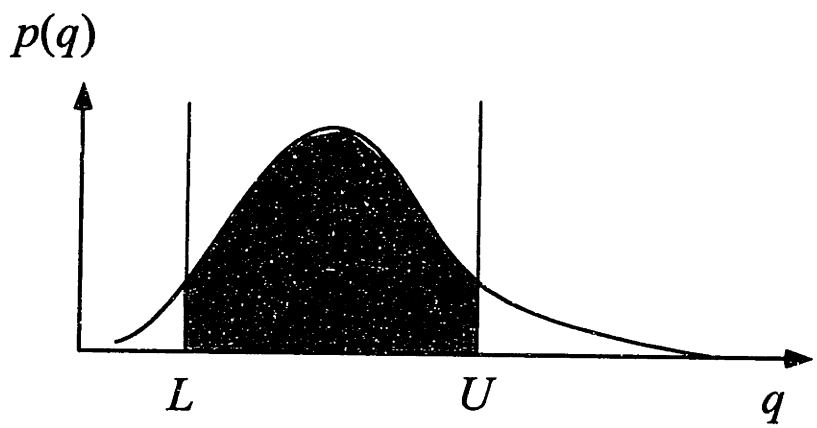


Figure 1.2.3. First time yield as an integral of probability density.

If the probability density function $p(q)$ over a given quality characteristic is Gaussian, then there exists an analytical expression for first time yield as a function of the process capability index and bias factor

$$Y_{FT} = \frac{1}{2}\left[\mathrm{erf}\left(\frac{3\sqrt{2}}{2}C_p(1-k)\right) + \mathrm{erf}\left(\frac{3\sqrt{2}}{2}C_p(1+k)\right)\right]. \tag{1.3.2.3}$$

Equation 1.3.2.3 above is derived in Section 5.3. It is interesting to note that the performance factor enters in the dominant term of Equation 1.3.2.3. The performance index does indeed determine one aspect of system performance – the first time yield.

Equation 1.3.2.3 is a reasonable predictor of yield if the data from the production process is nearly Gaussian. For the process of placing MCMs, we had $C_p = 0.82$ and $k = 0.08$. Plugging into Equation 1.3.2.3 gives an estimate of first time yield of 98.4%. The sample of data including measurements of 3,312 individual leads plotted in Figure 1.3.1.2 had a first time yield of 98.3%. The state of the art techniques seem to work well for computing first time yields. However, to design manufacturing systems, we need to go beyond first time yields.

1.3.3 Rolled Throughput Yield

As discussed in Section 1.2.1, it is not conformance to individual acceptance criteria that determines product acceptance. Engineering specifications generally require *simultaneous* conformance to *all* criteria. This inherent dominance in product acceptance criteria motivated the developers of six sigma producibility analysis to define *rolled throughput yield* (Y_{RT}). Rolled throughput yield is defined as the probability that every quality characteristic of an individual product is simultaneously met.

Rolled throughput yield is a more appropriate measure of system performance than first time yield. For example, consider the placement of multichip modules. The system performance metric we might wish to compute is rate of rework. What percentage of MCMs placed by this system will have to be sheared from the board, repaired, and re-soldered? The rework rate should be one minus the rolled throughput yield.

How can we make an estimate of rolled throughput yield without running expensive pilot production runs? It would be useful to be able to estimate rolled throughput yield based on data on first time yields. In the previous section, we noted that the MCM placement process has a first time yield of 98.3% based on experimental data. Given that each board has multiple leads, we know that rolled throughput yield will be lower than the first time yield, but how much lower?

In six sigma producibility analysis, it is suggested that rolled throughput yields be computed by multiplying the individual first time yields (Harry and Lawson, 1992).

$$Y_{RT} \approx \prod_{i=1}^{m} Y_{FTi} \,. \qquad (1.3.3.1)$$

This formula is simple and easy to apply. However, it assumes statistical independence among the quality characteristics of the products. In section 1.2.2, we saw that correlation among quality characteristics is common. Making an unwarranted assumption to get a nice analytical solution is like dropping your keys on one side of a street and going to the other side to look for them because the light is better*.

How badly does the unwarranted assumption of statistical independence affect the estimate of rolled throughput yield? In the case of the placement of the MCMs, Equation 1.3.3.1 predicts a rolled throughput yield of 0.983^{368} or 0.2%. The physical interpretation of this calculation is that only two MCMs in one thousand will pass inspection. In the experiments, six of out of nine of the MCMs passed inspection. The formula offered in six sigma producibility analysis gives an estimate of rolled throughput yield in error by *over two orders of magnitude.*

Manufacturing engineers need to be able to make reasonable predictions of rolled throughput yield so that they can discriminate among competing system designs. Correlation is the rule rather than the exception in manufacture and can significantly affect rolled throughput yield. Analytical techniques that relax the assumption of statistical independence among quality characteristics are clearly needed.

1.3.4 Quality Loss

As defined by ANSI, product acceptance is binary; Either a part is accurate enough or it is defective. Taguchi has encouraged engineers to rethink this "goal post" mentality. He correctly points out that the distinction between a part slightly outside the tolerance range and a part slightly within the range is arbitrary. If we think of the value we assign to a part as a function of its dimensions, engineering tolerances effectively assign a discontinuous function to this value (Fig.1.3.4.1)

As an alternative to the "goal post" mentality, Taguchi proposes a quadratic *quality loss function.* For plus and minus tolerances, it is defined as

$$L(q) = \frac{A_o}{\left[(U-L)/2\right]^2}\left(q - \frac{U+L}{2}\right)^2. \qquad (1.3.4.1)$$

* I first heard this analogy in a lecture by Stan Gershwin.

where A_0 is the cost to scrap or rework the product, q is the dimension or other quality characteristic of the product, and U and L are the upper and lower tolerance limits.

The quality loss function is motivated by the fact that it is a simple, continuous function that meets the boundary conditions that quality loss must be zero when the dimension equals the basic dimension and equals A_0 at the upper and lower tolerance limits. Taguchi's quality loss function encourages decision makers to recognize that products that are closer to the basic dimensions usually perform better and create more value for the customer. Thus, the quality loss function can be a useful tool for evaluating performance of a manufacturing system.

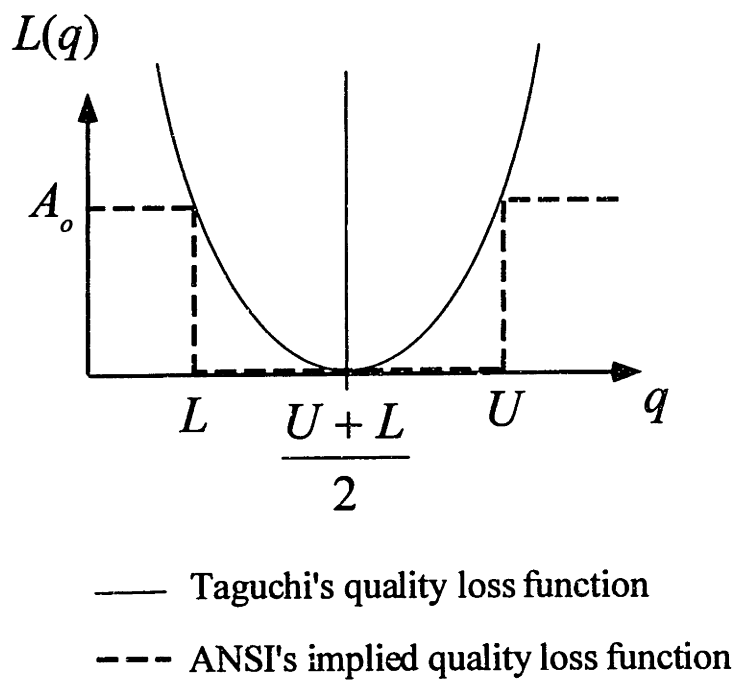


Figure 1.3.4.1 Quality loss functions as defined by Taguchi and implied by ANSI.

Engineers should exercise some caution in using Taguchi's quality loss function. Sometimes the loss function is presented as a universal law of nature derivable by expanding the Taylor's series about a point. In fact, it is just a convenient local approximation. It has a number of properties that may misguide decision making. For example, the Taguchi quality loss function continues to rise precipitously above A_0 outside the tolerance range. It is not clear what physical meaning to attach to a "quality loss" that exceeds the loss due to scrap. If a defect lies far outside the tolerances, it is likely to be detected and scrapped. It therefore seems reasonable that the loss function should be bounded above by A_0.

Because of these concerns, this thesis most often uses yield (rather than quality loss) as the critical performance metric to drive manufacturing system design. Quality loss can be a useful measure for manufacturing engineers to consider in system design. It should, in my opinion, only be used with caution.

1.4 My Approach

We have considered the nature of the problem faced by manufacturing system designers and surveyed some of the available tools. Now, I discuss my overall approach to meeting the goal of this thesis.

1.4.1 Mathematics versus Heuristics and Simulation

The goal of this thesis is to help engineers design manufacturing systems whose products conform to their dimensional tolerances. With this top level goal in mind, I now state my top level solution strategy. I will pursue an approach based on mathematical modeling. It is my opinion that, wherever a mathematical approach is possible, it is to be preferred over heuristics or simulation based approaches.

Let us consider an example of a heuristic approach. In a text on Design for Manufacture (DFM), one author offers the rule "products should be designed so that there are no adjustments required in assembly" [Anderson, 1990]. This rule has been abstracted from case studies that show that, in many cases, eliminating adjustments improved quality and reduced cycle time. The problem with this heuristic, as with all heuristics, is that it fails to identify its own limitations. There are plenty of cases in which assembly adjustments are justified and the rule offers no guidance in identifying these exceptions. This thesis offers a rigorous alternative to the above stated rule. It can identify both the cases in which adjustments should be removed, and the cases in which the adjustments are effective. Further, the approach defined here unambiguously states the conditions under which it applies and *proves* that the results hold under those conditions.

Let us now consider an example of a simulation based approach. Statistical assembly modeling is a popular approach to assembly tolerance design. For example, Variation Simulation Analysis Inc. offers popular software called VSA [Craig, 1988]. The user defines an set of components, probability density functions for dimensions of the components, an assembly procedure, and a set of dimensions to measure. The software samples from the appropriate

density functions, creates the assemblies, and measures and stores the appropriate dimensions. This procedure will give you correct answers if done properly. However, such a statistical assembly model is not the simplest possible representation of the problem. This thesis will show that the all the relevant information concerning most statistical assembly problems can be captured in a matrix/vector pair. The advantage of this parsimonious description is that it enables compact theorems that provide insight into system performance and design. As a corollary to Ockham's razor, I submit that the simplest representation of a problem enables the greatest depth of insight.

1.4.2 Probability versus Determinism

Dimensional variation is frequently modeled as random, but it almost invariably can be traced to some cause given sufficient time and resources. Bryan [1984] has claimed that this deterministic nature of manufacturing variation holds in most cases down to the level of nanometers and possibly to tenths of nanometers. Bryan concludes that use of probability and statistics to make manufacturing decisions should be viewed with caution. His views are summarized in the following excerpt from his paper entitled "The Power of Deterministic Thinking in Machine Tool Accuracy" [Bryan, 1984]:

> A determinist will never agree that a fixed value of non-repeatability can be assigned to a given machine. Such a value does not exist. Non-repeatability depends primarily on the time, money, and skill (culture) of the user.
> A determinist will be happy to agree that some level of apparent non-repeatability does exist for a given machine, on a given day, for a given series of tests, programmed in a given way, conducted by a given person, in a given environment consisting of a given level of temperature variation, vibration, and dirt, using a given set of instruments with a given limitation of time and money.
> A determinist might also agree that the assumption of a Gaussian distribution and the calculation of sigma values, although technically invalid, might be useful as a statement of the above variables existing at the time of a test.
> A determinist will have great anguish in ever agreeing that some level of non-repeatability is inevitable regardless of the time, money, and skill available. This issue is negotiable, however.

My approach to the problem of variation in manufacture is strongly influenced by this line of thinking. In this thesis, I assume that any given machine behaves deterministically. However, as Bryan notes, for a given machine under a prescribed set of conditions, there is some degree of apparent non-repeatability. Given that this is the case, it is an economic necessity to manage and tolerate some degree of variation.

My approach is to make a modeling assumption that incorporates both statistical and deterministic approaches. I assume that the dimensional variation observed in products is determined completely by underlying *noise factors* but treat these as noise factors as random variables. I treat the system behavior as deterministic and the input signal as probabilistic. My approach recognizes the underlying determinism of manufacture yet applies probability as an aid to decision making.

1.4.3 Refining the Solution Strategy

Given the preference for a mathematical approach, and the decision to use probability to support design decision making, and the multi-input multi-output nature of manufacture, my top level solution strategy is multivariate probability calculus. Having chosen a solution for my top level requirement, I may now define sub-requirements (Figure 1.4.3.1)[*].

To support manufacturing system design, I require a parsimonious description of manufacturing systems along with some means to map that description into predictions of relevant system performance metrics. A *parsimonious* description is desired to minimize the modeling effort required to make good decisions. A mapping into performance metrics is required to provide a rational basis for design decisions.

To satisfy the need for a parsimonious description of manufacture, I propose that all the important characteristics of any given manufacturing process with respect to quality and variation can be captured by modeling manufacture as an affine transformation of a random vector. As discussed in Section 1.2, there are key aspects of manufacturing systems that must be captured by any complete representation of a manufacturing system – variance, correlation, bias, and system architecture. Variance and correlation may be lumped under the term "covariance" and considered as one major facet of manufacturing systems. Covariance is captured in the capability matrices defined in this thesis. Similarly, bias will be represented by a bias vector. System architecture will be represented by using these matrices and vector in system block diagrams.

[*] The zig-zag mapping approach to problem solving was introduced by Suh [1990].

As indicated in Figure 1.4.1, I seek means to map the mathematical representation of manufacturing systems into performance metrics. Much of this thesis is concerned with expanding this branch of the solution. This thesis will develop a suite of equations, theorems, and algorithms that allow engineers to make inferences about system performance based on the representation of systems as capability matrices, bias vectors, and block diagrams.

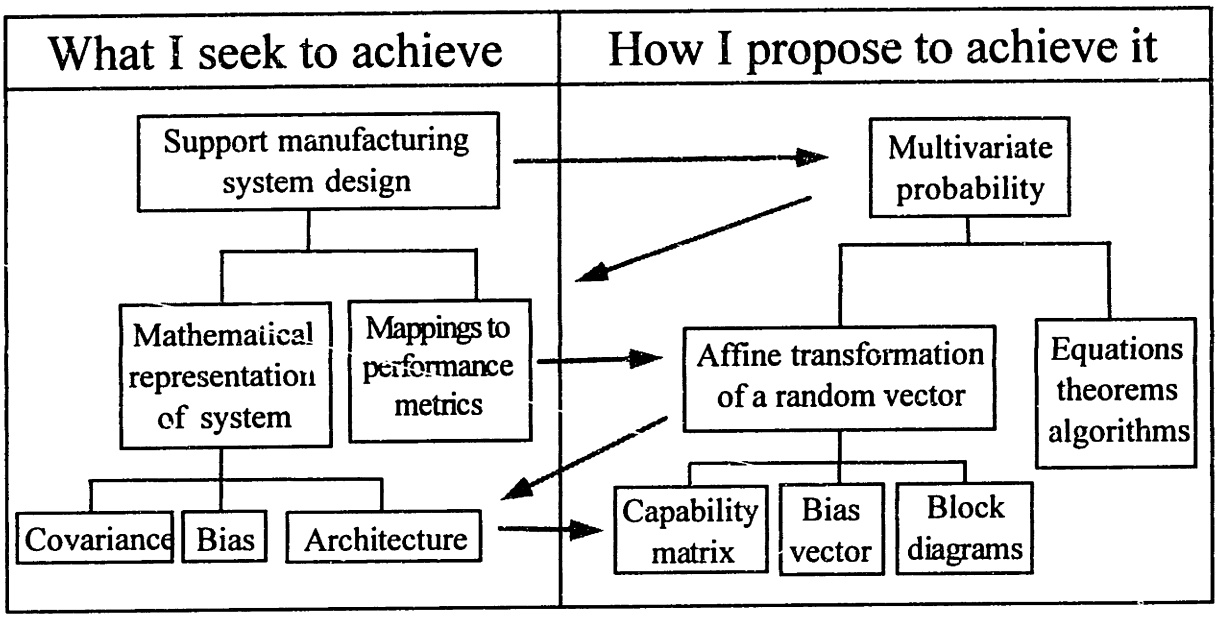


Figure 1.4.3.1. My solution strategy.

1.5 Guide to this Thesis

This document is tightly integrated; Most chapters build upon preceding chapters. This can make it difficult to read a subset of the document. To help readers browse through and extract only the information they want, I attempted to map the dependencies among the chapters in Figure 1.5.1. A line connecting a chapter with a preceding chapter denotes such a dependency. The reader may find it difficult to visit any node without first visiting the parent node. To decide which nodes are worth the trip, you may wish to read through the brief chapter summaries below:

Chapter 2 introduces the primary case study of the thesis – surface mount of multichip modules. It comes first because it is used as a practical example in Chapters 2 through 8. Each time a new concept is introduced, I tried to show how it applies in the context of surface mount processes.

Chapter 3 defines the central concept of the thesis, the process capability matrix and bias vector. It motivates their definition based on traditional scalar capability indices and explains how to construct capability matrices for manufacturing processes and systems.

Chapter 4 introduces an important application of capability matrices – their use in block diagram representations of manufacturing systems. In this section models of individual processing steps are combined and reduced to models of entire systems.

Chapter 5 develops ways to use the capability matrix to estimate rolled throughput yield. It considers special cases such as Gaussian and uniform noise. It also describes more general procedures for estimating yield based on Monte Carlo integration.

Chapter 6 explores the effect of tolerances and specifications on rolled throughput yield. It introduces qualitative means to identify the best tolerances to widen and the best noise factors to reduce.

Chapter 7 introduces a powerful use of manufacturing system block diagrams – their use in modeling on-line adjustment procedures. Generalized inverses are used to define feedforward control strategies guaranteed to minimize quality loss. Theorems define the conditions under which quality loss can theoretically be eliminated.

Chapter 8 adapts the ideas of Chapter 5 to design of product inspection procedures. The chapter introduces a novel definition of key characteristics based on conditional probability.

Chapter 9 describes a method for modeling machining processes. This chapter can be read independent of the rest of the thesis. It is, however, a prerequisite for fully understanding the case studies in Chapters 10 and 11.

Chapter 10 presents a case study of a form grinding machine for engine poppet valves. The machine is shown to exhibit significantly non-linear behavior over the region of interest. The case study exposes a limitation of the capability matrix approach to manufacturing system analysis.

Chapter 11 presents a case study of a grinding machine for engine crank shafts. The virtual machining framework is shown to be predictive of machine behavior and a valuable diagnostic tool. The form tolerances on the crank pins are shown to be a strongly non-linear function of noises in the system. Nevertheless, the capability matrix approach proves useful for error budgeting.

Chapter 12 considers related work. It appears at the end of the thesis so that I can discuss the relationship of this thesis to earlier work without foreshadowing all the major results. It can be read after the introduction, but I do not recommend it.

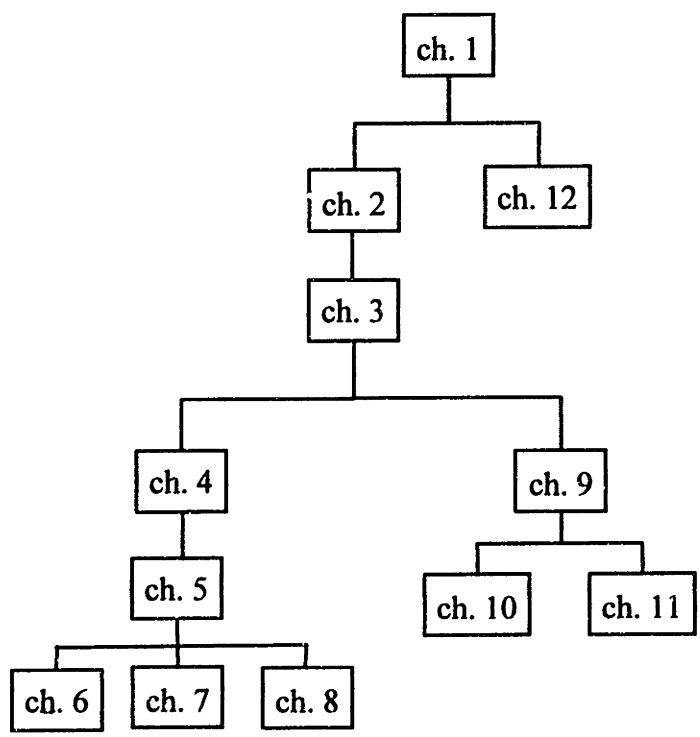


Figure 1.5.1. A dependency map among the chapters of this thesis.

2. Surface Mount of Multichip Modules

This chapter introduces the primary case study in this thesis – surface mount of multichip modules. The case study is weaved throughout the thesis to illustrate the practical application of theoretical results. This chapter serves to put the case study in an appropriate context, to explain the function of the product, and give an overview of the manufacturing process.

2.1 Electronics Manufacture and Economic Competition

This section argues that production system design is especially important in electronics packaging. This makes the surface mount of MCMs an excellent case study for illustrating the system design tools developed in this thesis.

Hans Danielsson, in a recent text on electronics manufacture [Danielsson, 1995], makes several key observations about the competitive playing field in electronics manufacture:

- Electronics manufacture is highly automated. Only about 10% of total production cost is associated with the cost of labor.
- The electronic components and materials are a commodity. Every company, no matter what part of the world it operates in, pays roughly the same prices for components.
- The processing equipment is a commodity. The production equipment available to firms throughout the world is strikingly uniform.

If companies around the world are using similar equipment to process components with the same price structure in highly automated processes, how can one company gain competitive advantage over another? No company can get much benefit from low labor costs or a more skilled manual labor force. No company has exclusive access to significantly better process technology or equipment. In effect, every company that wishes to manufacture electronic products must create its production system from the same kit of building blocks as its competitors.

If every company must choose building blocks from the common pool of process equipment and components, how can it gain competitive advantage? The answer lies in better system design. Each company has design freedom in the particular building blocks it chooses, the way it

arranges the blocks into an integrated whole, and the way it applies process control and inspection protocols over the arrangement of blocks. If a company designs its production systems better than its competition, it will achieve higher yields at lower cost even though it had access to the same kit of components. System design is the key to competitiveness in electronics manufacture.

In the case study in this dissertation, an electronics manufacturing company is charged with manufacturing *units* that perform a function in a military application. The units are composed of *slices* which are the vertically aligned structures evident in the Figure 2.1.1. The slices are, in turn, composed of printed wiring assemblies populated with electronic components. To be competitive, the company must design a production system which can manufacture these units with high yield at a reasonable cost. The company will create a competitive advantage by designing and managing this system better than other defense electronics companies.

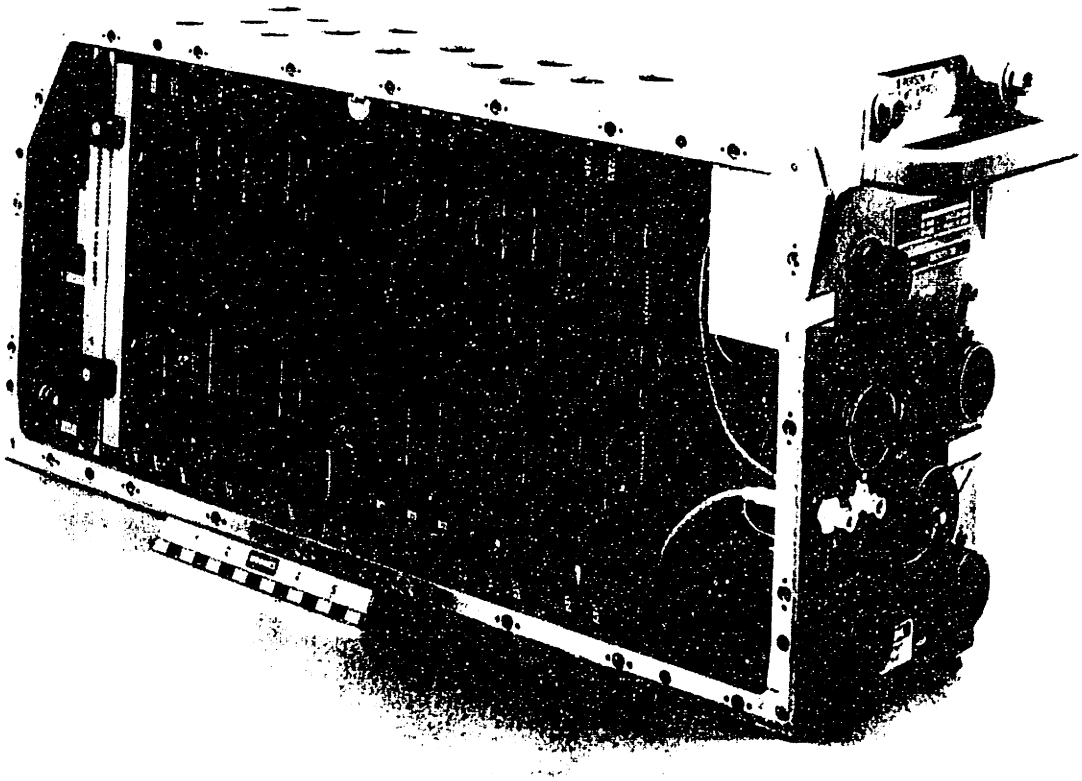


Figure 2.1.1 An electronics unit for some unspecified military application.

2.2 Electronics Packaging and Multichip Modules

Electronics manufacture can be considered as having two distinct steps, the manufacture of semiconductor devices, and the packaging of these devices. The case study in this thesis concerns the latter. This section will provide a brief overview of electronics packaging concerns and discuss the structure of multichip modules.

An electronics package is that which interconnects, supports, protects, powers, and cools semiconductor devices. Electronic packages are not just passive containers for silicon; The design of the package has a significant effect on overall system performance and reliability.

At present, electronic packaging represents a substantial performance bottleneck. For the past thirty years, silicon transistor speed and on-chip density have grown exponentially. Chip interconnection speed and IO count have not grown nearly as fast. At the time of publication, semi-conductor performance is measured in pico-seconds while the performance of computers is measured in nanoseconds. The three order of magnitude difference between on-chip and off-chip performance is attributable to losses in the package [Doane and Franzon, 1993]. Thus, most gains in computer performance will likely come from improvements in electronics packaging.

Multichip modules (MCMs) are one of the most promising solutions to opening up the performance bottleneck of electronics packaging. MCMs have several advantages over other packaging technologies. MCMs have the higher packaging efficiency than most packaging alternatives. Typically, greater than 30% of the area of in an MCM is composed of silicon ICs. This results in increased system speed and reduced overall size for a given application. MCMs achieve these advantages with their unique structure.

An MCM is a structure consisting of two or more integrated circuit (IC) chips electrically connected to a common circuit base and interconnected by conductors in that base [Doane and Franzon, 1993]. This allows a tight grouping of ICs in a protected module. The structure of an MCM is shown schematically in Figure 2.2.1.

The common circuit base provides signal interconnection circuits, power and ground distribution, and mechanical support. The electrical connections between the ICs and the common circuit base are known as *first level connections*. First level connections are usually made by wire bonding. The electrical connections between the common circuit base and the PWB are known as *second level connections*. The second level connections are usually made by

ribbon leads brazed to the common circuit base and soldered to the PWB. In this thesis, I will be concerned with the effect of manufacturing variation on second level connections. The processes used to create the second level connections are referred to collectively as *surface mount technology* and are the focus of the next section.

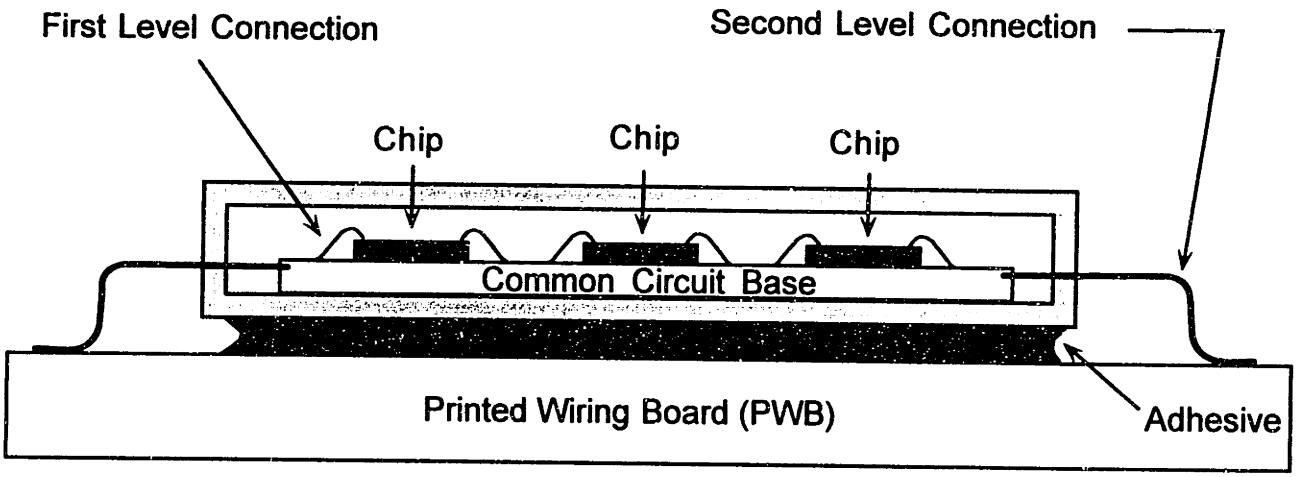


Figure 2.2.1 A schematic diagram of an MCM's structure.

2.3 Overview of Surface Mount Technology

The second level connections between the multichip module and the printed wiring board are made by a process that falls under the broad category of surface mount technology. This section will provide a brief introduction to the field.

In 1943, Paul Eisler invented the printed circuit board (PCB), a two dimensional pattern of copper foil attached to an insulating phenolic paper board. The PCB quickly became the dominant means of interconnecting electronic components. The basic concept of using a printed two dimensional structure to interconnect electronics is still dominant today [Strauss, 1994].

The initial means of attachment of a component to a PCB was by plated through holes. Wherever a component was to be attached to the PCB, a hole was drilled and surrounded by a ring of conductive foil. The leads on the component were inserted through the hole and soldered.

As the pin count of components increased, it became essential to dispense with insertion of wires into drilled holes. This led to the growth of surface mounted devices which are soldered

directly to pads on the surface of the PCB. In about 1993, surface mount devices overtook inserted components in number sold per year [Strauss, 1994].

Surface mount technology refers to the processes required to attach electronic components (mechanically, thermally, and electrically) to the surface of a printed circuit board (PCB). Surface mount technology includes processes for fabricating PCBs, stenciling solder paste on PCBs, fabricating surface mount components, tinning leads, attaching components, and soldering.

Variation can enter into any of the processing steps in the surface mount process. The PCB itself may be wavy, the solder paste may be stenciled too thickly or in the incorrect position or shape, the soldering time and temperature may vary. However, a study of quality of the solder joints in quad flat packs has shown that mounting position of the semiconductor package is perhaps the most significant factor in industrial practice [Ogata and Takei, 1987]. Given this evidence and the greater size of the MCM packages, this thesis will focus on component placement and other errors that affect lead position.

Many different types of components can be surface mounted. Many passive components require only two connections to the PCB. They generally have electrically conductive caps which solder to the PCB. Larger, application specific integrated circuits (ASICs) and multichip modules (MCMs) require large numbers of interconnections to the PCB. There are many different structures used to provide these interconnections including arrays of pins, balls, or pads. The MCM in this case study has a frame of ribbon leads.

2.4 Lead Frames and Lead Forming

The MCMs in the case study are electrically connected to the printed circuit board by ribbon leads. The manufacturing facility modeled here is supplied with electronic components with an unformed lead frame (Fig. 2.4.1). Lead frames are typically stamped from a cold rolled sheet of Alloy 42 (42% Nickel/ 58% Iron) [Strauss, 1994]. This alloy has a low coefficient of thermal expansion which reduces the mechanical stresses on thermal cycling. The frames are usually 0.2 to 0.3 mm thick. This thickness is adequate to ensure the leads will not plastically deform during normal handling. The lead frame is, however, thin enough to allow some mechanical deflection. This is needed to absorb the strains due to coefficient of thermal expansion mismatch between

the component and PWA. Ceramic tie bars serve to protect the leads and maintain even spacing until the lead forming process is complete. The lead frame is brazed to the common circuit base.

The MCM placed in the case study has 368 leads. The longer sides (sides one and three) have 125 leads each while the shorter sides (sides two and four) have 59 leads each. The leads are numbered consecutively counterclockwise around the periphery of the MCM (see Figure 2.4.1).

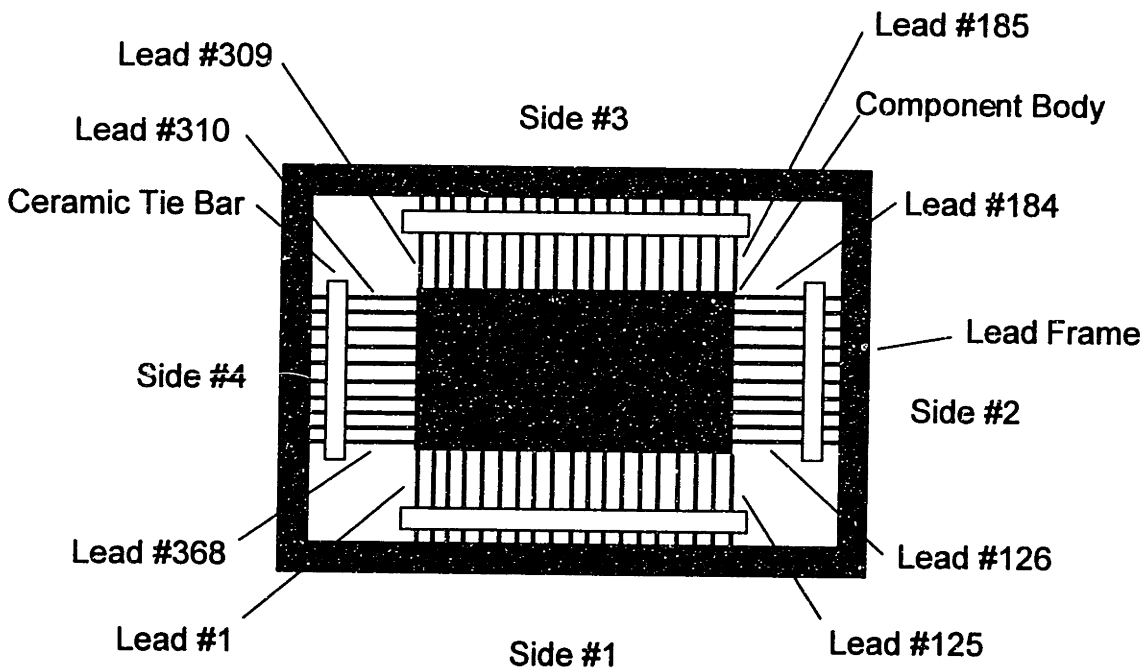


Figure 3.4.1 A 368 leaded MCM as delivered to the production facility.

Prior to surface mount of the MCM, it is necessary to create a desired *bend group*, a ribbon lead geometry specified for a particular slice design. The bend groups define the dimensions of a leg shape. Much of the terminology used to describe the leg geometry is anatomical in origin. The horizontal portion of the lead closest to the body is referred to as the *thigh*. The thigh is connected by a roughly 90 degree bend, or *knee*, to a vertical portion, or *shin*. The shin is attached to the *foot* through a *heel* bend.

The leads extend below the component body to create an *air gap* between the bottom of the package and any flat surface on which it rests. The air gap often provides space for adhesives. If

there is no air gap, there will likely be open solder joints. There also tend to be open solder joints if the feet do not have sufficient coplanarity (all the feet should lie in the same plane).

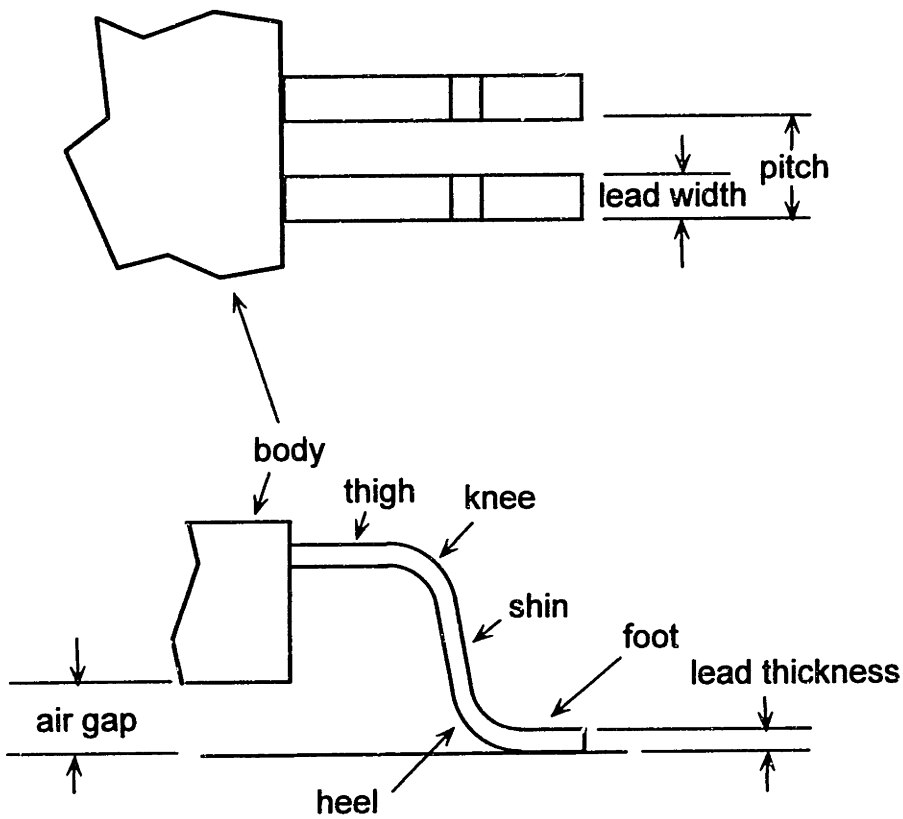


Figure 2.4.2 Formed ribbon lead shape and terminology.

The leads are formed into the leg shape on automated systems (Fig 2.4.3). The components are aligned in fixtures by touching off on the edge of the component body. The components are then moved by a robotic arm from the alignment fixture to the fixed forming dies. Alternately, the skilled technician in Figure 2.4.3 may manually place the component in the forming die.

The operation of the lead forming die is depicted in Figure 2.4.4. The MCM rests on a floating anvil which allows the lead frame to rest on the die edge (Fig. 2.4.4a). The upper face of the die moves down to create the leg shape (Fig. 2.4.4b). A shear trims the excess lead frame from the feet (Fig. 2.4.4c). When the upper face of the die retracts, the legs will spring back to some extent at both the knee and heel bends (Fig. 2.4.4d).

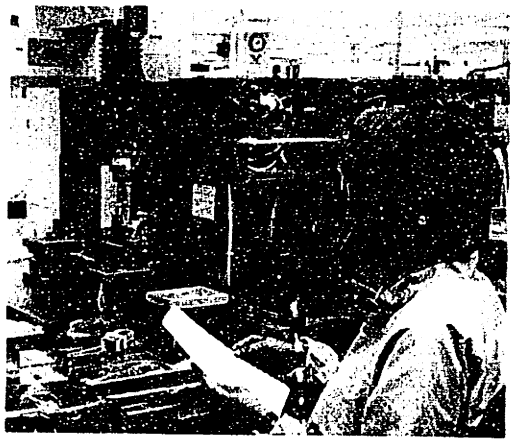
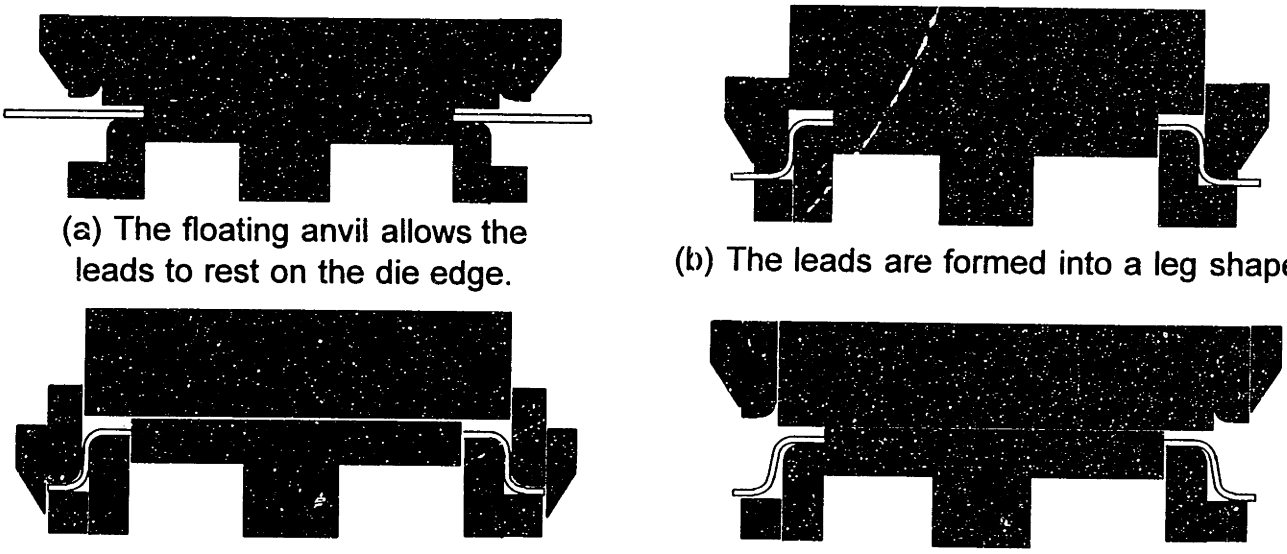


Figure 2.4.3 An automated system for lead forming.



(a) The floating anvil allows the leads to rest on the die edge.

(b) The leads are formed into a leg shape.

(c) A shear trims the excess lead.

(d) The knee and heel spring back.

Figure 2.4.4 The lead forming process.

There are many sources of variation introduced in the process of manufacturing and forming the leads. Some problems are evident in the lead frame as delivered to the production facility. There tends to be variation in the scale of the lead frame. In other words, the pitch of the leads is sometimes greater or less than nominal. Also, there tends to be some variation of each ribbon lead from its desired position. In other words, the pitch is not uniform from lead to lead.

There may be some misalignment of the lead frame on the MCM body. There may also be misalignment of the MCM in the lead forming die. Both of these factors cause a regular pattern

of error in the foot positions. The angular misalignment causes a displacement of the foot proportional to the perpendicular distance to the center of the component. Superposed on this error is an additional component of foot position error due to vertical misalignment of the shin. As the lead wraps around the knee bend, the angular misalignment in the plane of the lead frame becomes a vertical misalignment (Fig 2.4.5). This causes a lateral drift of the foot approximately proportional to the product of the misalignment angle and the nominal length of the shin.

The pattern of error is evident upon close inspection of Figure 2.4.6. Here a quad flat pack was intentionally misaligned within a lead forming die. The vertical misalignment of the shins can be detected in the front view. The top view reveals that the thigh length varies linearly along each side of the component.

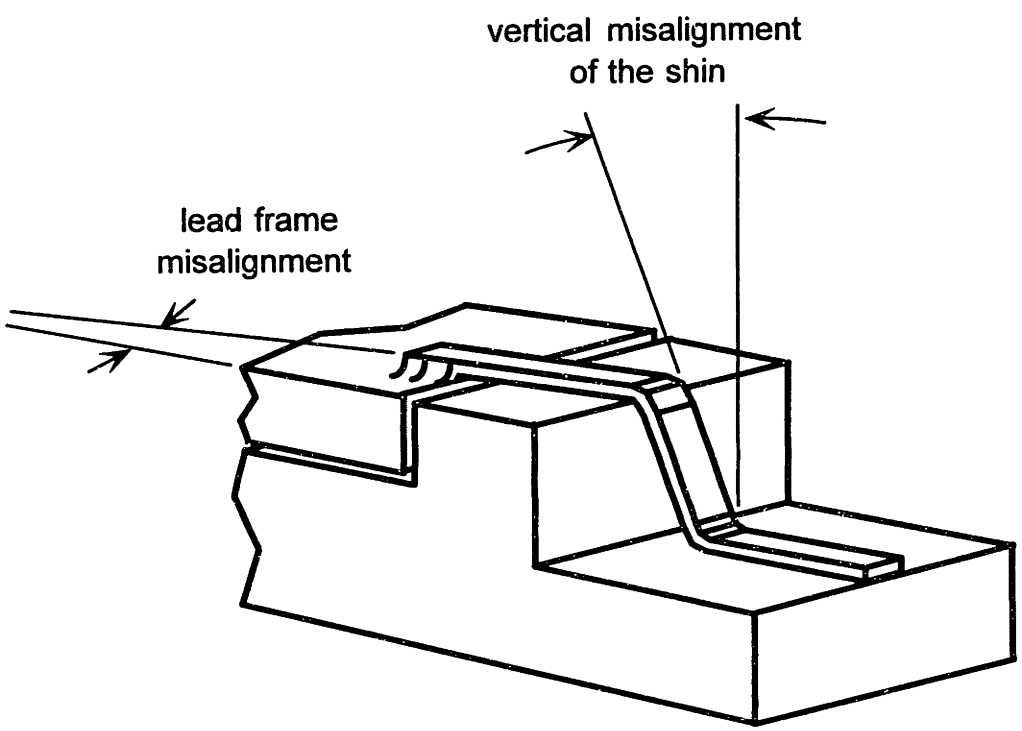


Figure 2.4.5 Vertical misalignment of the shin caused by lead frame misalignment.

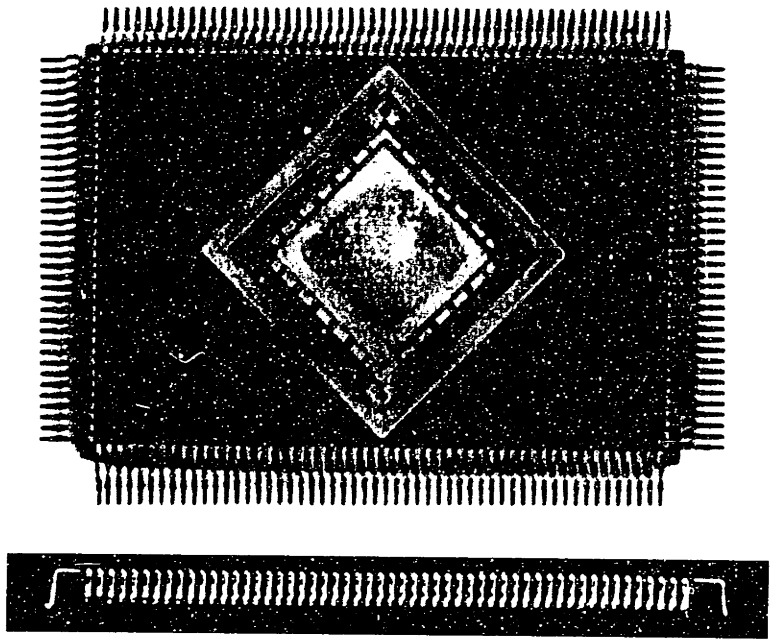


Figure 2.4.6 Pattern of error due to misalignment of a quad flat pack in a lead forming die.

Spring back may cause quality problems especially if it is not repeatable. The springback in a sheet wrapped around a mandrel of radius R_i is given approximately by

$$\frac{R_i}{R_f} = 4\left(\frac{R_i Y}{ET}\right)^3 - 3\left(\frac{R_i Y}{ET}\right) + 1 \tag{2.4.1}$$

where R_f is the final radius of the formed sheet, T is the sheet thickness, E is the Young's modulus of the sheet, and Y is the yield strength of the sheet [Kalpakjian, 1995]. Therefore, the degree of springback in the knee and heel is a function of the thickness of the leads, and the stiffness and yield strength of the lead material. Because the lead frame is cold rolled, the material may not be isotropic and the leads may spring back by different amounts on different sides of the MCM.

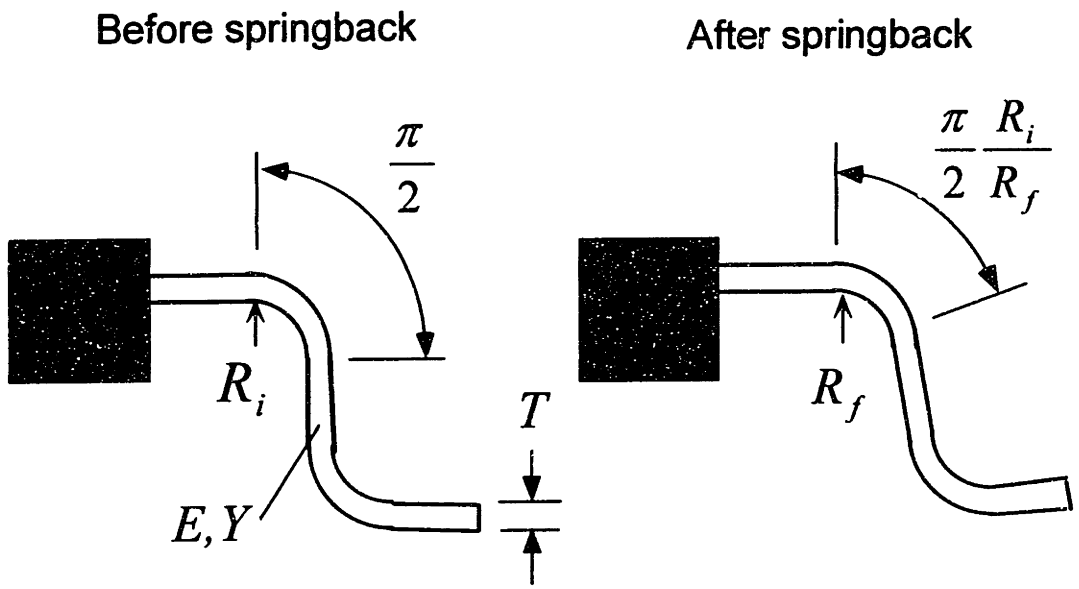


Figure 2.4.7 Springback of a ribbon lead during a forming operation.

This section has briefly reviewed the process by which the leads on the MCM are manufactured, attached, and formed. A more detailed analysis of lead forming processes is provided in Appendix B. After the leads are formed, they are tinned by dipping them in a solder wave. The rest of the manufacturing operations to be considered here are performed on a robotic assembly system introduced in the next section.

2.5 The Robotic Assembly System

After the leads of the component have been formed and tinned, the component is moved to a robotic assembly system. The assembly system in our case study is based on commercially available, gantry style, four axis positioning system called the Adept UltraOne. The base is granite. The rails for the x and y axes are ceramic and act as air bearing surfaces. In the case study, the machine has been specially fitted with palates, fixtures, an array of soldering heads, and an adhesive dispense system.

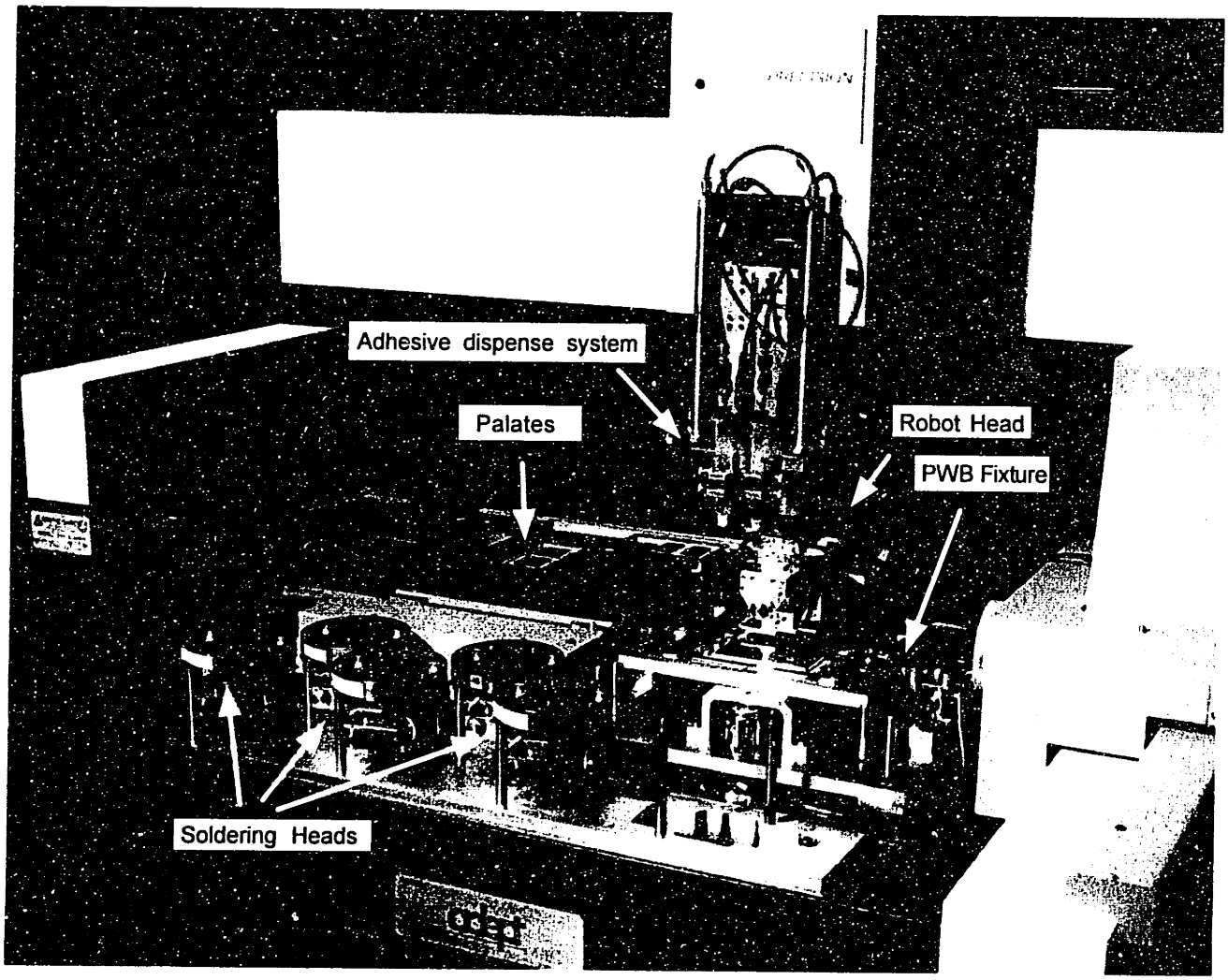


Figure 2.5.1 The robotic assembly system.

2.6 Robotic Inspection and Adjustment

This section introduces the machine vision system which is incorporated in the assembly robot. The vision system serves to inspect the leads on each component. The data from the inspection is used to determine an optimal placement of the leads on the pads. The data is also used to determine if the lead positions after adjustment meet minimum criteria for lead accuracy.

As discussed in section 2.4, there are several means by which variations can be introduced in the foot positions. Some of the variations in these foot positions may be compensated by employing the available degrees of freedom of the assembly robot. In this assembly system, the angular orientation and the x and y position of the MCM may be used to partially compensate for

variations introduced upstream. Figure 2.6.1 illustrates the process. For clarity, only the corner leads are shown. On the left is an MCM placed in its nominal position. If the leads and pads were in their nominal positions, the leads and pads would be perfectly aligned. However, due to the variations introduced in lead forming, et cetera, the leads are poorly aligned with the pads. The robot has the freedom to place the component at some position and orientation other than the nominal position. It is possible that some adjustment of the MCM away from nominal may lead to better alignment of the leads with the pads as shown in the right side of Figure 2.6.1.

Computing the optimal position is straightforward. Let us define an optimal adjustment as that set of x, y, and angular orientation values that minimizes the sum squared deviations of the distances from the leads to the pads. Given this definition of optimality, the values of the free adjustment parameters can be computed by solving for the least squares solution of a linear system of equations using the measured positions of the leads. Optimal adjustments will be discussed in greater detail in Chapter 8.

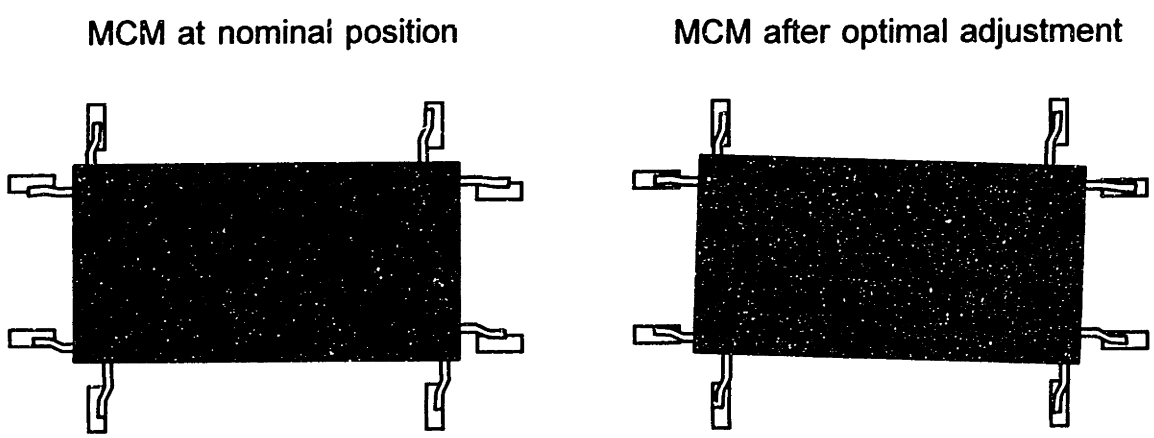


Figure 2.6.1 The effects of the optimal adjustment procedure.

Not every MCM will be compensated adequately by an optimal adjustment procedure. Some will have errors that are too great or have errors that do not lend themselves to compensation by such an adjustment. For example, if the corner lead of an MCM is deformed during handling, it is unlikely that an adjustment will correct the problem. Any repositioning scheme that brings the corner lead onto the pad will drive the other leads off their pads.

Therefore, after the optimal adjustment is computed, the robotic assembly system must determine if the component will meet a set of acceptance criteria.

In defense electronics, the acceptance criteria for lead alignment with pads may be defined by MIL-STD 2000A. The military standard sets several specifications on the position and orientation of the leads with respect to the pads. For example, section MIL STD 2000 A 4.23.7.2 defines limits on the distance that the lead can extend past the edge of the pads (Fig. 2.6.2) (the figure uses the term "land" which is synonymous with "pad" in this context). The specification states "side overhang is permissible provided it does not exceed 25 percent of the lead width or ... 0.02 inch, whichever is greater."

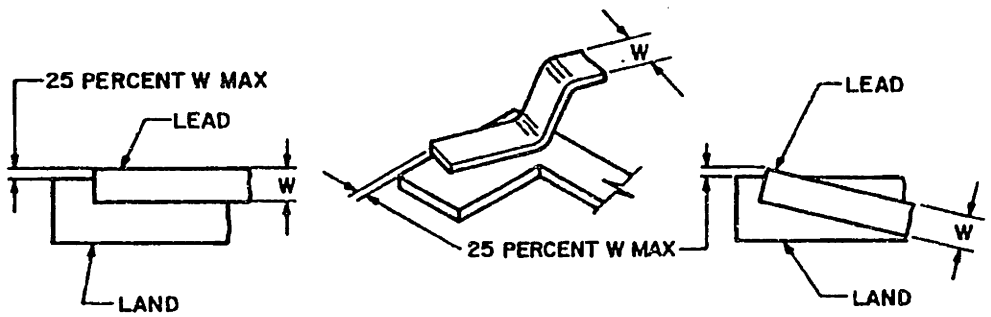


Figure 2.6.2 Military standard tolerance for foot side overhang.

There are also specifications set on toe and heel overhang (Figure 2.6.3). Toe end overhang is permissible "provided the total overhang does not exceed 25 percent of the lead width ... or 0.02 inch, whichever is less." No overhang of the heel is permitted over the land (Fig. 2.6.4). This is due to the fact that the heel fillet created during solder reflow is critical to solder joint life.

If the assembly robot determines that *any* of the leads on the MCM will fail to meet the standards even given that the proper adjustment is made, then the MCM must be reworked. If the MCM passes this test, The robot will continue with the assembly process. The next step in the process is adhesive application as described in the following section.

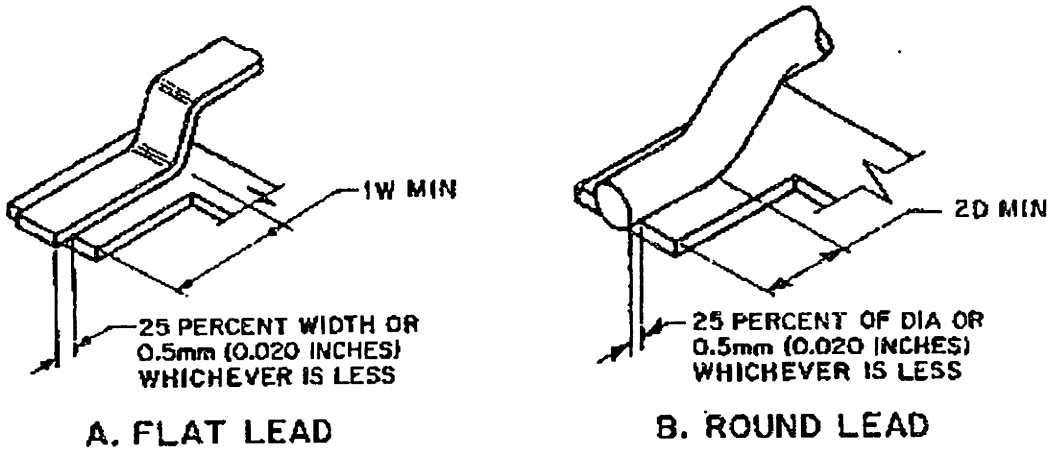


Figure 2.6.3 Military standard tolerance for foot end overhang.

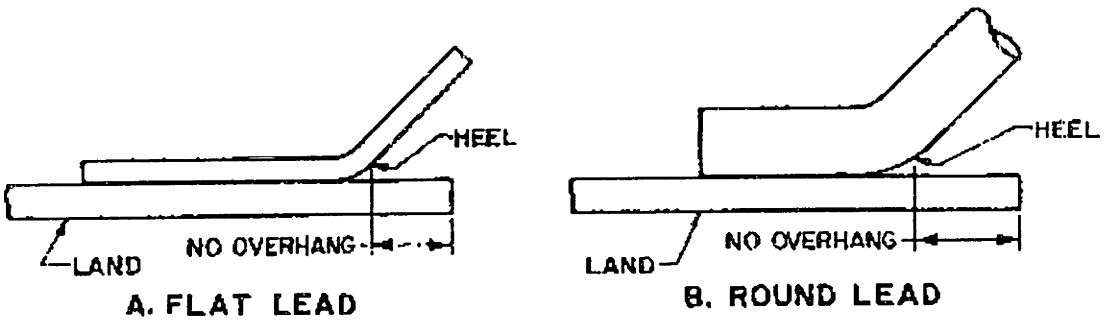


Figure 2.6.4 Military standard tolerance for heel overhang.

2.7 Adhesive Application

Before the assembly robot places most components, it applies adhesives to the site of attachment. Adhesives provide mechanical support for the MCM. Adhesives are often filled with ceramic particles to improve heat transfer from the MCM to the PWB. The PWB is often laminated on an aluminum heat sink which absorbs and dissipates the heat generated in the components. In some cases the adhesives are electrically conductive to provide a ground for the electronic components.

The adhesives are usually thermosets and come in premixed tubes which are refrigerated prior to use. The adhesives are expelled through a nozzle by air pressure as the robot moves the tube at constant velocity (Figure 2.7.1). This creates a bead of adhesive. Since the MCMs are

much wider than the adhesive bead and high percent coverage is desired, the robot traces out a serpentine pattern as it lays down a bead (Figure 2.7.2). The robot then places the MCM on the serpentine pattern and applies a downward force according to a set schedule. The downward force on the MCM causes the beads to flatten as the adhesive flows outboard (Figure 2.7.3).

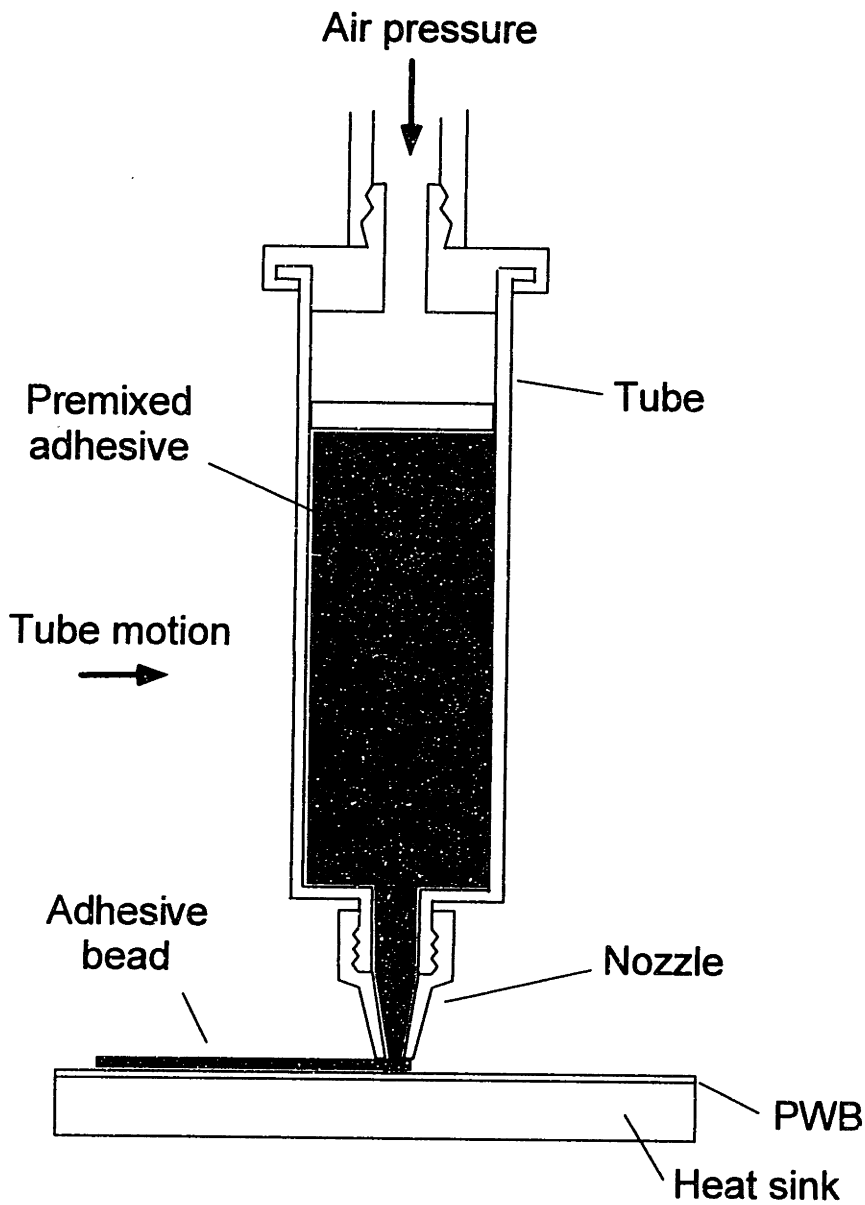


Figure 2.7.1 The adhesive dispense system.

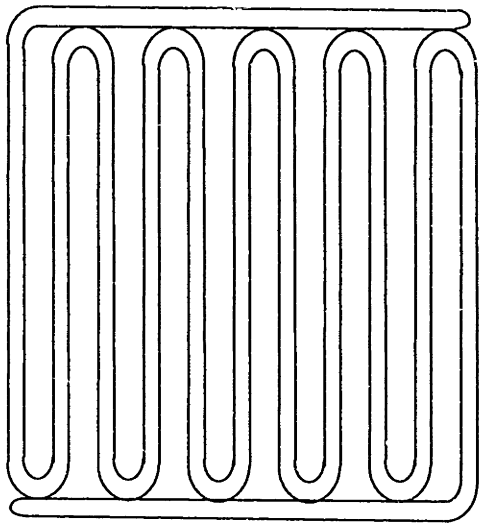


Figure 2.7.2 A serpentine pattern of adhesive.

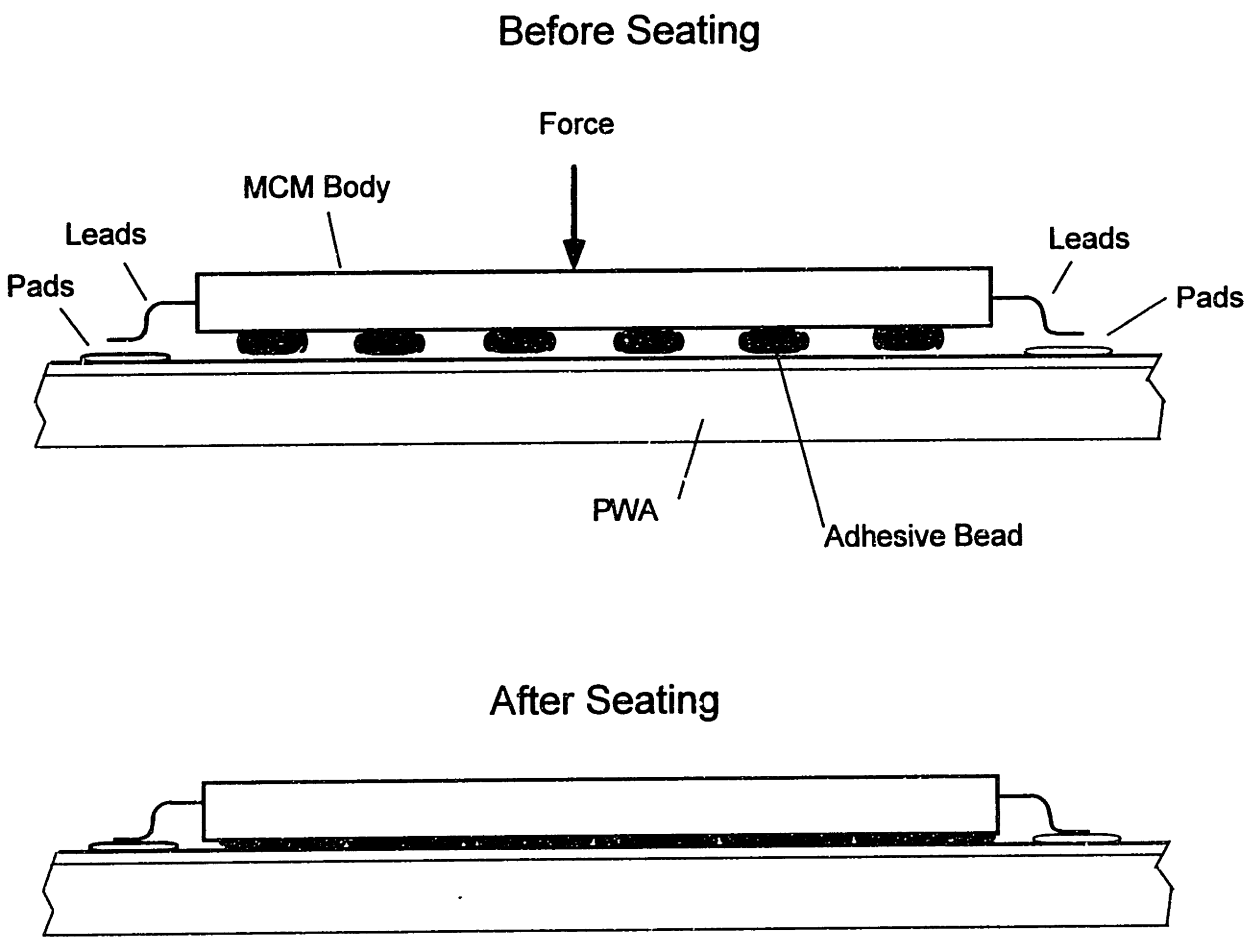


Figure 2.7.3 Seating the MCM on the adhesive.

The outboard flow of adhesive can be modeled as plane Poiseuille flow. Figure 2.7.4 is a schematic diagram of the flow conditions and the terminology used in the model. In the simplest case in which the bead is assumed to be much longer than it is wide and the effects of air pressure are neglected, the final height of the chip h is related to the time t over which downward force is applied

$$h(t) = \left[\frac{5F/L}{\mu W_o^3 h_o^3} t + \frac{1}{h_o^5} \right]^{-1/5} \tag{2.7.1}$$

where F/L is the downward force per unit length of the bead, μ is the viscosity of the adhesive, W_O is the original width of the adhesive bead, and h_O is the height of the component above the PWA. The development of this equation and analysis of more complex flow conditions are documented in Appendix C.

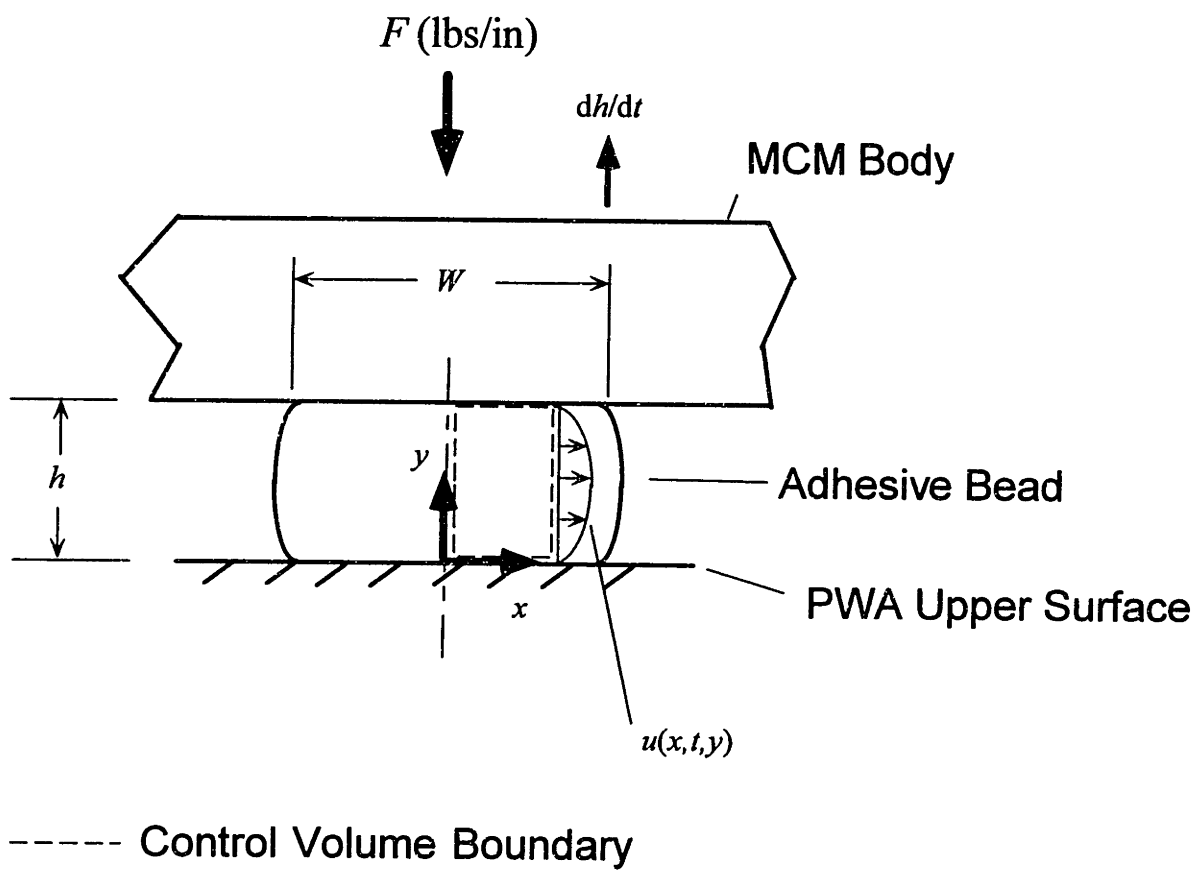


Figure 2.7.4 Modeling outboard flow of adhesive under a seating force.

There are several sources of variation in the adhesive application process. The viscosity of the adhesive may vary with temperature, with time as it cures, and batch to batch due to composition and handling conditions. The viscosity of the adhesive affects the amount of adhesive applied and also the amount by which the component seats under downward force. There may also be air bubbles in the adhesive tubes. This will cause breaks in the serpentine pattern of adhesive and affect percent coverage.

The variations that enter in the adhesive application and seating process can ultimately affect the solder quality. For example, if the component seats too far, the downward force intended to seat the component may deform the leads. This phenomenon has been dubbed "the dead spider effect" due to the way the feet point upward (Figure 2.7.5). The dead spider effect will reduce the contact area between the foot and the pad and may cause weak or insufficient solder joints. It is also possible for adhesive to flow out from under the component and contaminate the solder joints.

After the adhesive has been applied and the component has been seated, the leads are soldered to the pads. This process is discussed in the next section.

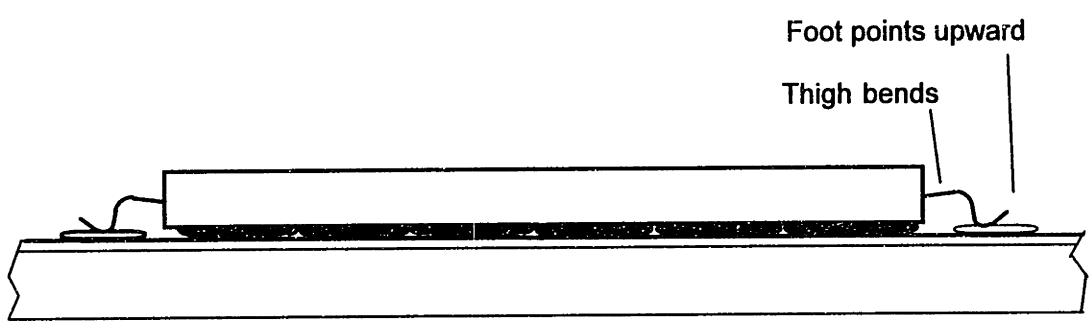


Figure 2.7.5 The "dead spider effect" caused by excessive seating of the component.

2.8 Hot Bar Soldering

There are several ways to attach a surface mounted device (SMD) onto a PCB. Some components are wire bonded directly to the PCB, some are attached via electrically conductive adhesives, but the vast majority are soldered. There are many means of soldering. Some

components are placed on pads of solder paste and soldered en mass by heating in a reflow oven. Some components are soldered individually by computer controlled lasers that are rastered over their surfaces. The MCM in this case study has its leads soldered by a hot bar soldering process.

In the hot bar soldering process, many leads are soldered at one time. A bar shaped, thermostatically controlled, resistive heating element is pressed down upon a bank of leads. The leads are heated above the solder reflow temperature so that the solder will flow. The solder must flow under the forces of surface tension and gravity to fill the areas required to make a good electrical connection and a mechanically sound joint. After the temperature has been raised above the reflow temperature for a set period of time, the current is shut off. The bar normally is held in place until it cools below the solder freezing temperature. The bar may then be lifted free of the completed joints.

The hot bar solder process has several advantages. The fact that it applies a downward force on the leads can compensate for errors in coplanarity of the leads. Also, the soldering is performed under conditions of near thermodynamic equilibrium. Thus, the process is not a sensitive to changes in solder quantity, heat transfer rates, etc. Hot bar soldering therefore usually will not result in burned leads due to excessive temperatures or in open joints due to inadequate temperature.

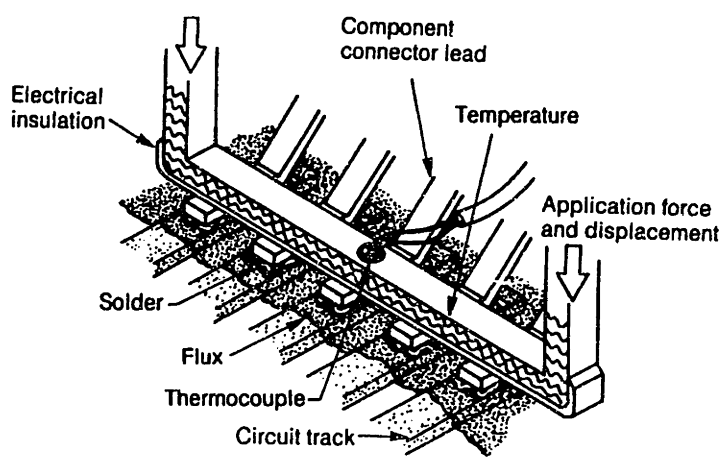


Figure 2.8.1 The hot bar soldering process [Danielsson, 1995]

2.9 Experimental Procedure

To study the surface mount process, a small set of experimental data was collected. Nine 368 leaded MCMs with unformed leads were purchased. The leads of these MCMs were formed and tinned. The MCMs were then mounted on PWAs by the Adept UltraOne robot. The results of the machine vision system's measurements of the lead positions were recorded. Based on these measurements, the side-to-side error in the lead positions were computed for two possible placement strategies. One placement strategy was to set the MCM at the nominal location and orientation. The other placement strategy was to employ the least squares adjustment procedure described in section 2.6. The data set therefore includes nine sets of 368 measurements or 3,312 total data points which are transformed into two different sets of possible system outputs.

All of the data presented in this thesis has been scaled to mask the capabilities of the system from which the experimental data was derived. Any specific numerical figures concerning tolerance widths, mean shifts, and standard deviations have been altered from the actual values. This masking changed the number of nonconforming leads and number of nonconforming MCMs the system produced. However, the masking procedure preserved the correlation coefficients of the data set.

An example of a masked data set is provided in Figures 2.9.1 through 4. All nine masked data sets are presented in Appendix D. All the data in these figures is normalized with respect to a tolerance width that differs from the actual tolerance width of the system.

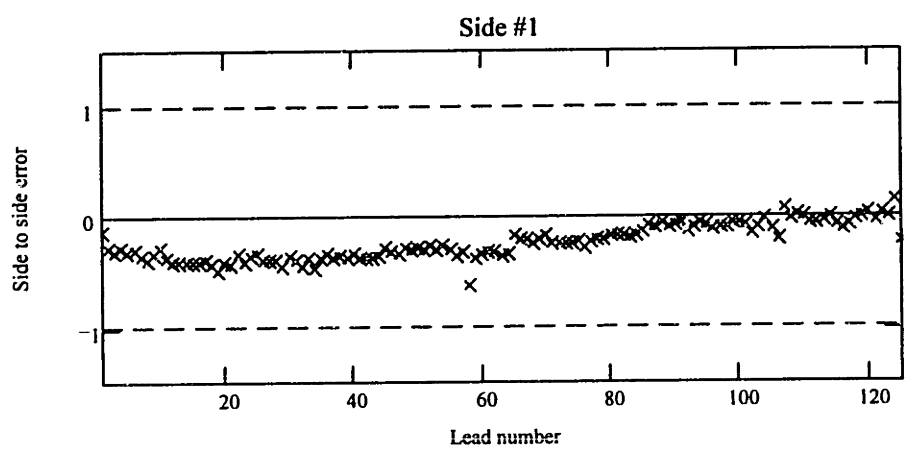


Figure 2.9.1 Data from the surface mount of an MCM – Side #1.

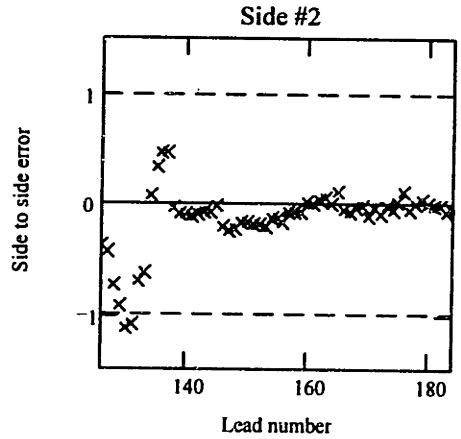


Figure 2.9.2 Data from the surface mount of an MCM – Side #2.

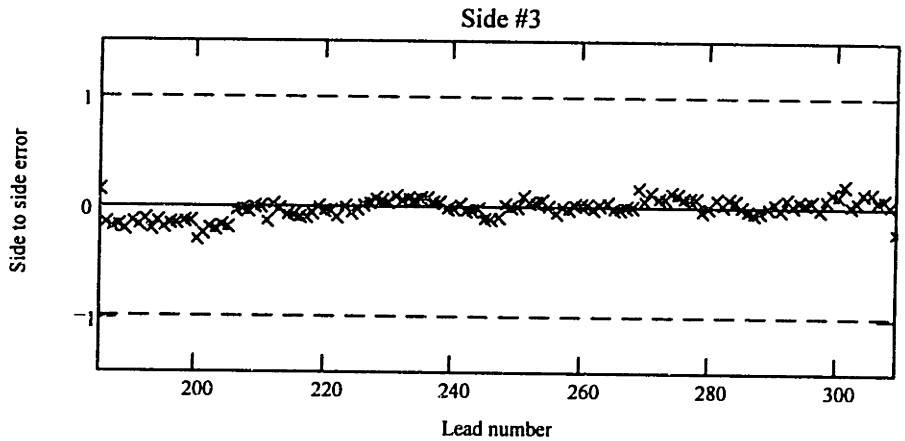


Figure 2.9.3 Data from the surface mount of an MCM – Side #3.

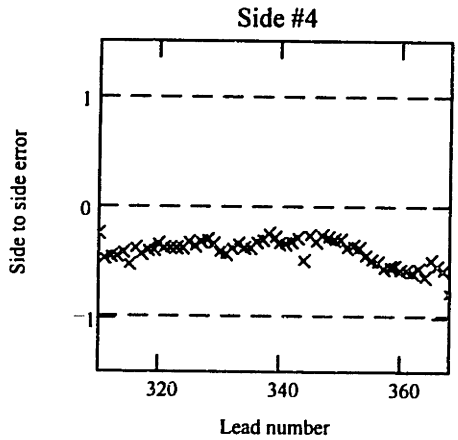


Figure 2.9.4 Data from the surface mount of an MCM – Side #4.

2.10 Case Study Overview

This chapter has provided an overview of the case study that will be threaded throughout this thesis. The case study concerns the process of assembling electronic components called multichip modules (MCMs) onto printed wiring boards (PWBs). This process is of considerable industrial importance given the size of the electronics industry and the effect of packaging on system performance and reliability.

The process is summarized in the schematic diagram of Figure 2.10.1. The manufacturing system under consideration processes incoming material including MCMs with flat lead frames, printed wiring assemblies (PWAs), and tubes of adhesive. The processing is performed by lead forming dies, wave soldering machines, and a robotic assembly station. The assembly station measures the foot positions on the leads, dispenses adhesive on the site of MCM attachment, adjusts the position of the MCM, seats the MCM on the adhesive, and solders the component in place with a hot bar. The final product of the process is a slice populated with electronic components.

The process of surface mount of MCMs exemplifies many of the characteristics of modern manufacturing processes. It has multiple outputs in the form of 368 lead positions. It has dominance among the outputs since a single short or open connection will cause the slice to malfunction. It has multiple inputs given the many noise factors that can enter the process such as lead frame misalignment and robot accuracy. It has multiple processing steps that propagate variation from one step to another. It incorporates adjustment variables by measuring variations in lead positions and attempting to compensation with MCM position and orientation. Thus, the MCM surface mount process described in this section will serve as a useful example in all of the theoretical developments to follow in chapters 3 through 8.

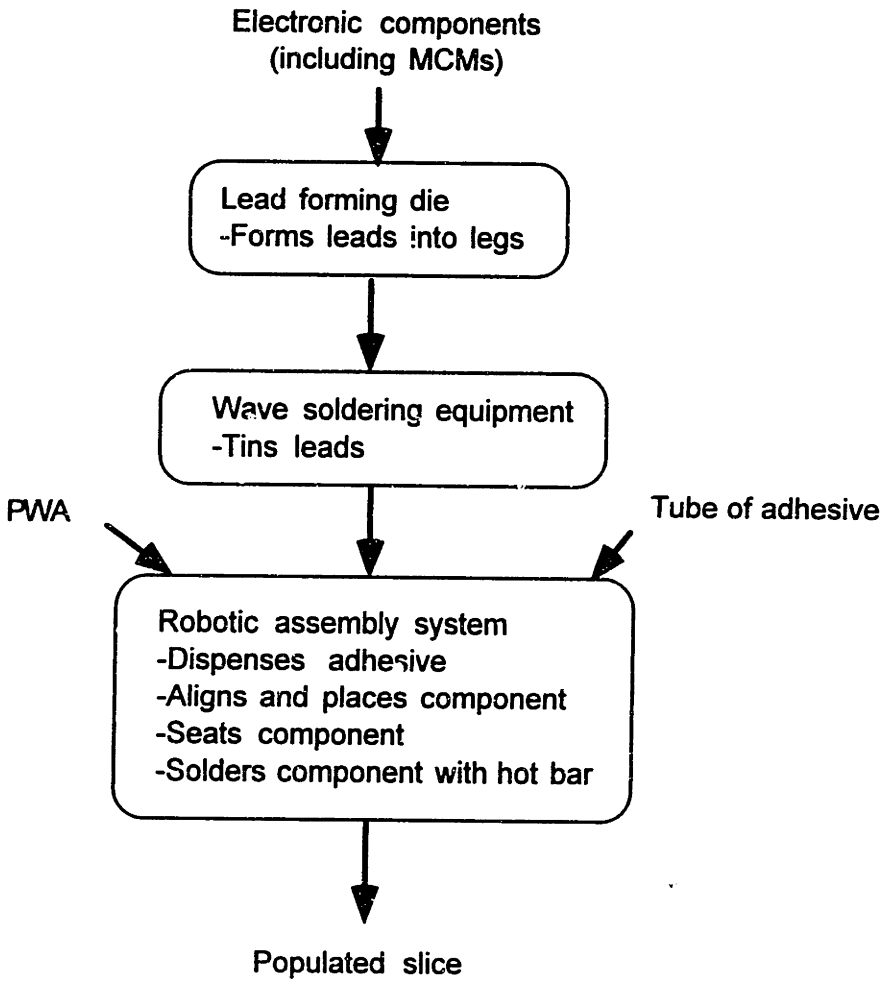


Figure 2.10.1 Overview of the MCM surface mount process.

3. The Process Capability Matrix

The purpose of this chapter is to introduce a mathematical representation of manufacturing systems called a process capability matrix. The matrix captures information on variation, sensitivity, tolerance, and correlation in an ordered set of dimensionless coefficients. The chapter will motivate the approach, define the capability matrix, compare the capability matrix to capability indices, and develop capability matrices for the surface mount of MCMs.

3.1 Quality Characteristics

In the field of robust design, the term *quality characteristic* is defined as *the measured response of a design* [Fowles and Creveling, 1995]. I wish to adapt this term to the goal of this thesis. The goal is to help engineers design manufacturing systems whose products conform to their dimensional tolerances. In this context, the *design* which is responding is that of the manufacturing system*. Further, the *measured response* of the manufacturing system is the set of dimensions of *discrete units of product*. In manufacture, *acceptance criteria* for discrete units of product are defined by tolerance limits placed on the values of the dimensions. This leads us to a definition of quality characteristics suited to design of manufacturing systems.

Definition The *quality characteristics* of a discrete unit of product are the complete set of dimensions whose tolerances define the product's acceptance criteria.

Quality characteristics are often geometrical dimensions such as size, location, and form. Product performance metrics may also be quality characteristics. As long as the product acceptance criteria define some measurable quantity and places limits its value, that quantity is a quality characteristic of that product.

3.2 The Quality Vector

In order to form a mathematical representation of manufacture, I will collect all the quality characteristics of a discrete unit of product into a vector $\mathbf{q}$. Therefore, $\mathbf{q}_i$ refers to the ith quality

* In the spirit of simulteneous engineering, one might define the system as being both the prodcut and process. However, to maintain focus, I will assume that the product design is frozen.

characteristic. Let m be the number of quality characteristics. The vector $\mathbf{q}$ will therefore have m elements.

Definition A *quality vector* is an ordered set of quality characteristics from a discrete unit of product.

Quality vectors exhibit variation even under "apparently equal" operating conditions of the manufacturing system. Therefore, I will model the quality vector $\mathbf{q}$ as a random vector. The manufacture of a discrete unit of product under the prescribed "apparently equal" operating conditions is, in the parlance of probability theory, an *experiment*. An *instance* of the random vector $\mathbf{q}$ is the outcome of a single *trial* of the experiment. Each instance of $\mathbf{q}$ is associated with a discrete unit of product.

The acceptance criteria apply to instances of quality vectors and define an event – *conformance*. Conformance is determined by tolerance limits on each quality characteristic. For the purposes of this thesis, I will assume that there is always both an upper and lower limit on each quality characteristic. We may define two vectors L and U that contain the upper and lower tolerance values. The limits on the quality characteristic $\mathbf{q}_i$ are L_i and U_i.

Most product dimensions are continuous valued. Therefore, the elements of the quality vector $\mathbf{q}_i$ are modeled as continuous random variables. Any set of continuous random variables has an associated joint density function. I will denote the joint density function over the quality characteristics as $p_q(\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_m)$.

3.3 The Quality Vector for Surface Mount of MCMs

To provide a feel for the meaning of quality characteristics and quality vectors in industrial practice, this section will construct the quality vector for the surface mount of MCMs.

The quality characteristics for the surface mount of MCMs are denoted $\mathbf{q}_i$ which is the difference between the actual and nominal locations of the ith foot centroid. The sense of the error is defined as positive where the foot is displaced *counterclockwise* as viewed from above (Fig. 3.3.1). For the purposes of this thesis, I will consider the half tolerance width $(U_i - L_i)/2$

to be 240μm for every quality characteristic ($\mathbf{q}_i$). This and other specific quantities have been altered to mask the capabilities of the system from which the experimental data was derived.

Table 3.2.1 Quality characteristics for surface mount of MCMs.

Index	L	U	Description
$\mathbf{q}_1$	-240μm	240 μm	error in side to side position of foot #1
$\mathbf{q}_2$	-240μm	240 μm	error in side to side position of foot #2
⋮	⋮	⋮	⋮
$\mathbf{q}_{368}$	-240μm	240μm	error in side to side position of foot #442

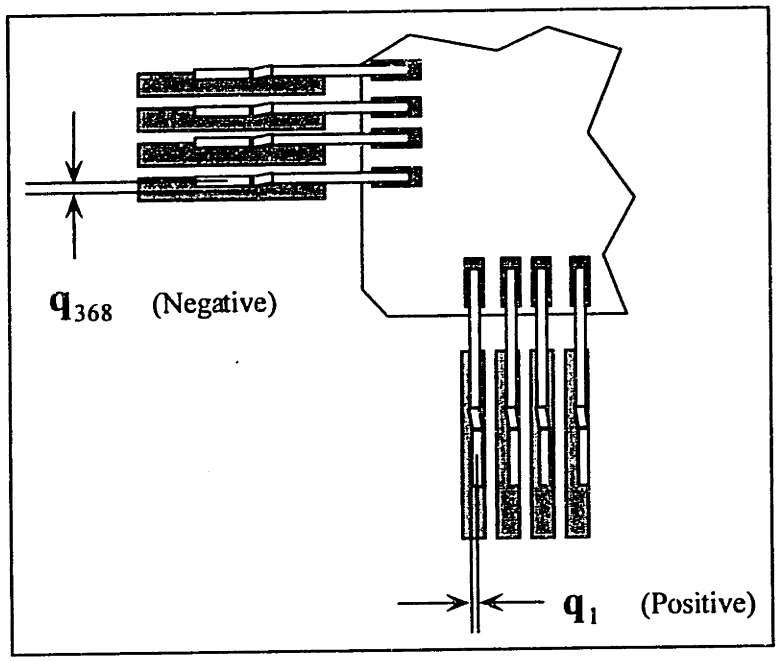


Figure 3.2.1 The quality characteristics for surface mount of MCMs.

3.4 Noise Factors

In the field of robust design, a *noise factor* is defined as *the cause of variation in the quality characteristics* [Fowles and Creveling, 1995]. The term "noise factor" can be adapted slightly to suit the purpose of this thesis.

Definition A *noise factor* of a manufacturing system is a quantity whose variation causes variation in the quality vector.

Noise factors may be associated with the manufacturing equipment, its environment, or the materials which the manufacturing system transforms. Noise factors in the manufacturing equipment include, for example, inaccuracies in machine components such as lead screw errors which affect machining accuracy. Noise factors in the environment include, for example, ambient temperature and humidity which affect adhesive flow. Noise factors in incoming materials include, for example, properties such as yield strength of sheet stock or accuracy of parts in an assembly operation.

3.5 The Noise Vector

In order to form a mathematical representation of manufacture, I will collect the noise factors into a noise vector $\mathbf{n}$. Thus $\mathbf{n}_j$ refers to the jth noise factor in a manufacturing system. Let n be the number of noise factors. The vector $\mathbf{n}$ will therefore have n elements.

Definition A *noise vector* is an ordered set of noise factors of a manufacturing system.

Noise vectors exhibit variation even under "apparently equal" operating conditions of the manufacturing system. Therefore, I will model the noise vector $\mathbf{n}$ as a random vector. Most noise factors are continuous valued. Therefore, the elements of the quality vector $\mathbf{n}_j$ are modeled as continuous random variables. I will denote the joint density function over the noise factors as $p_n(\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_n)$. Like the quality vector, noise vectors are outcomes of trials associated with discrete units of product.

The elements of the noise vectors have means and standard deviations. To create a uniform notation, I define a vector of means and a vector of standard deviations

$$\mu_j \equiv E(\mathbf{n}_j) \tag{3.5.1}$$

$$\sigma_j \equiv E\left[\left(\mathbf{n}_j - E(\mathbf{n}_j)\right)^2\right] \tag{3.5.2}$$

where $E(\bullet)$ denotes the expected value of a random variable (the glossary in Appendix A defines this and other terms).

In some cases, the means and standard deviations of noise factors may be computed from experimental data. For example, if the noise factors are related to dimensions of incoming

components, then the supplier may have historical data on the variation in the components. In other cases, it may be necessary to estimate the standard deviation based on specifications. For example, it is often reasonable to assume that a tolerance range on a purchased component represents six times standard deviation of the population of the components.

3.6 The Noise Vector for Surface Mount of MCMs

To illustrate the meaning of the noise vector in industrial practice, I will construct and characterize a noise vector for surface mount of MCMs. This entails listing possible noise factors based on engineering judgment and experience.

In the surface mount of MCMs, there are certain noise factors that suggest themselves immediately. The drawing of the unformed lead frame includes tolerances on each lead position. It is reasonable to assume, therefore, that there is some random variation in the location of each lead in the unformed lead frame. One may posit the existence of a separate noise factor for each lead that captures this unintended variation. Let us call these noise factors $\mathbf{n}_1$ through $\mathbf{n}_{368}$. The standard deviation of the noise was assumed to be one sixth the tolerance width. The bias in these noise factors was assumed to be zero. The values assumed for this case study are listed in Table 3.4.1.

Another noise factor in the lead frame was hypothesized based on experience. In surface mount of large multi-leaded components, there is often a linear drift in side to side position of the leads on each side of a component. This linear drift is evident in the data set as shown in Figure 3.6.1. A physically reasonable cause of this drift is that the lead frame is larger or smaller than it should be. In other words, the pitch of the lead frame doesn't match that of the pads on the PWB. This "scaling effect" of the lead frame on lead positions is depicted in Figure 3.6.2. Let us call this noise factor $\mathbf{n}_{369}$. The standard deviation and bias of the noise were inferred from the data set.

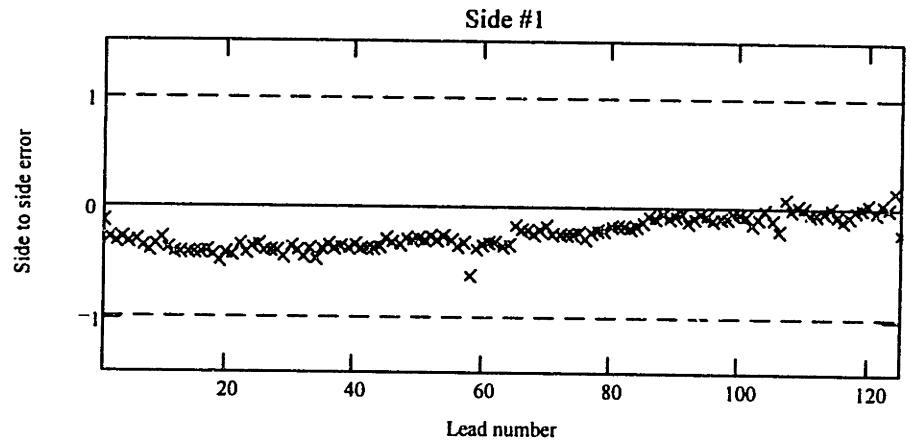
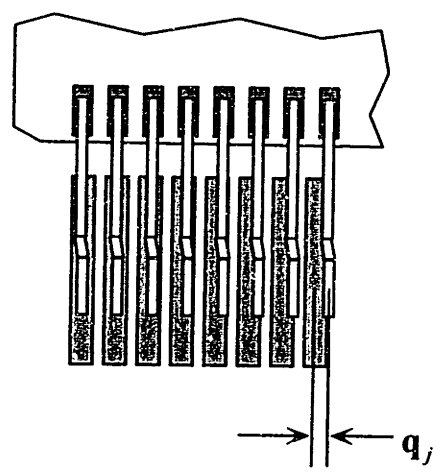


Figure 3.6.1 Data from the surface mount of an MCM – Side #1.



$\mathbf{q}_j \propto j$ for each side of the MCM

Figure 3.6.2 The effect of scaling of the lead frame.

After the lead frame is stamped, it is assembled onto the ceramic body of the MCM and brazed in place. There may exist some error in the x and y positions and in the orientation of the lead frame with respect to the body. These three noise factors are denoted as $\mathbf{n}_{370}$, $\mathbf{n}_{371}$, and $\mathbf{n}_{372}$ respectively and are listed in Table 3.4.1.

The assembly robot has limited accuracy and repeatability due to thermal errors, resolution of its angular encoders, and a host of other effects. The resulting errors in x and y position and in orientation of the MCM are denoted as $\mathbf{n}_{373}$, $\mathbf{n}_{374}$, and $\mathbf{n}_{375}$ respectively and are listed in Table

3.4.1. The bias and standard deviations can be estimated based on the robot manufacturer's specifications. The specifications have been altered for the purposes of this thesis.

Table 3.4.1 Noise factors for the electronics assembly example.

Index	μ	σ	Description
$\mathbf{n}_1$	0μm	29 μm	error in side to side position of lead #1
$\mathbf{n}_2$	0 μm	29 μm	error in side to side position of lead #2
⋮	⋮	⋮	⋮
$\mathbf{n}_{368}$	0 μm	29 μm	error in side to side position of lead #368
$\mathbf{n}_{369}$	0.01%	0.0001 %	error in scaling of the lead frame
$\mathbf{n}_{370}$	0 μrad	10μrad	misalignment of the lead frame on the MCM body
$\mathbf{n}_{371}$	0 μm	16.8 μm	x positioning error of the lead frame on the MCM
$\mathbf{n}_{372}$	0 μm	16.8 μm	y positioning error of the lead frame on the MCM
$\mathbf{n}_{373}$	0 μrad	3.33μrad	misalignment of the MCM on the PWA
$\mathbf{n}_{374}$	0 μm	8.4 μm	x positioning error of the MCM on the PWA
$\mathbf{n}_{375}$	0 μm	8.4 μm	y positioning error of the MCM on the PWA

3.7 Complete Noise Vectors

We have defined a noise factor as a quantity whose variation causes variation in the quality vector. As one expands the noise vector to include more of the underlying causes of variation, one should approach a complete model of the observed variation. To quantify this concept, let us define the concept of a *complete* noise vector.

Definition A noise vector is *complete* with respect to a product and its manufacturing system if, under the prescribed conditions of the experiment, there exists a deterministic mapping f such that

$$\mathbf{q} = f(\mathbf{n}) + \varepsilon$$

$$\text{where } E(\varepsilon_i) + 6\sqrt{E\left(\left[\varepsilon_i - E(\varepsilon_i)\right]^2\right)} \le \frac{U_i - L_i}{2} \text{ for all } i \in 1,\ 2,\ \ldots m \qquad (3.7.1)$$

The function f can be said to define the system's *response surface* with respect to the noise factors. The symbol ε denotes a vector of random variables that represent all the unmodeled error in the system. Equation 3.7.1 expresses an arbitrary limit on the amount of unmodeled error one will allow in an analysis effort and still deem it "complete". The limit was chosen to follow Motorola's definition of 6σ quality. That is, if one were to entirely eliminate the variation in the noise vector, then one would achieve 6σ quality in each quality characteristic.

It may be argued that, for reasonably complex manufacturing systems, it is impractical to find a complete noise vector – that there are too many possible contributors to variation. However, a "vital few" causes often account for most of the variation in quality characteristics (Juran, 1951, Bhote, 1991, Harry and Lawson, 1992). An effort to assign cause to variation can succeed if the vital few are included among a reasonably short list of candidate noise factors. Therefore Equation 3.7.1 does not imply a philosophical commitment to determinism nor does it demand exhaustive modeling of the manufacturing system. Rather, it is a mathematical expression of the Pareto principle as applied to manufacture.

Emanuel Sachs has noted that in many mature production processes, the Pareto principle seems to break down. Over time, the "vital few" causes of variation will be discovered and reduced. At the same time, there is economic incentive to relax specifications on noise factors that contribute little to variation. Therefore, with a long term effort, the process becomes "balanced"; A host of noise factors may contribute approximately equally to the observed variation in the process.

It may be difficult to define a complete noise vector for a carefully balanced system. Also, a balanced system may demand a very long noise vector making the modeling and computational efforts much greater. In these cases, the methods described in this thesis may be less valuable as a tool for error budgeting. On the other hand, a carefully balanced system has little margin for improvement by careful error budgeting in any case.

3.8 Manufacture as a Linear Transformation

As defined in section 3.7, there exists an approximate functional relationship between a complete noise vector and the quality vector. In order to proceed with an analysis of process capability, one must define the nature of that functional relationship. This section argues that an assumption of linear behavior will yield reasonably accurate results for most manufacturing systems.

The assumption of linear behavior requires justification since non-linear behavior is commonly observed in manufacture. For example, in the process of seating an electronic component, the relationship between chip height (a quality characteristic) and time of force application (a noise factor) is highly nonlinear (see Equation 2.7.1). It is important to recognize that this thesis does not assume *globally* linear behavior. Rather it assumes nearly linear behavior with respect to noise factors within a neighborhood.

The Taylor's series expansion of the system response f about a target vector $\mathbf{t}$ yields

$$\mathbf{q}_i = f(\mathbf{n})_i = f(\mathbf{t})_i + \sum_{j=1}^{n}\left[\left(\left.\frac{\partial \mathbf{q}}{\partial \mathbf{n}}\right|_{\mathbf{n}=\mathbf{t}}\right)_{ij} \cdot (\mathbf{n}_j - \mathbf{t}_j)\right] + \text{h.o.t.} \tag{3.8.1}$$

Figure 3.8.1 depicts the Taylor's series approximation of the system response about a target vector. If we define a region including $\pm 6\sigma_j$ about each of the noise factors, then the probability that any noise factor is outside the region is roughly $2n$ parts per billion if each noise factor is normally distributed. If the higher order terms in Equation 3.8.1 are sufficiently small in that region then the linear system model is adequate for nearly every product manufactured.

The target vector $\mathbf{t}$ is an arbitrary point at which the Jacobian of the response surface is evaluated. It is desired to make the higher order terms as small as possible throughout the neighborhood of interest. For that reason, the target vector should be selected so that it is close to the expected value of the noise vector.

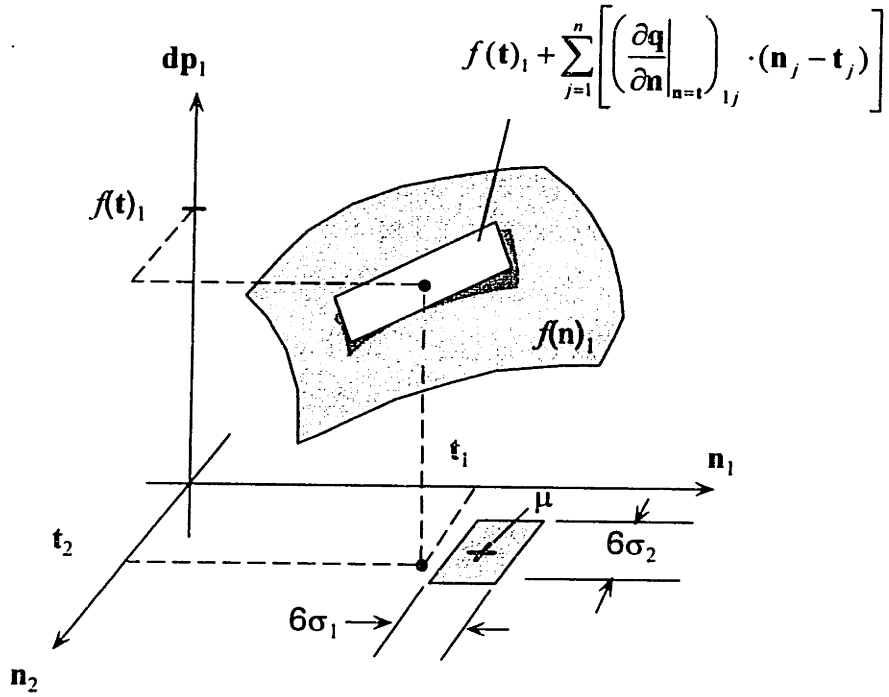


Figure 3.8.1 Linearization of the system response function.

The linearity assumption is supported by a survey of literature on parametric errors in machine tools and CMMs which found that higher order terms were rarely significant in parametric error models of a broad range of equipment (Soons, 1993). The reason for this phenomenon is that angular error motions tend to be small in equipment as precise as machine tools and CMMs. When error motions are small, kinematic models of these machines (see Chapter 9 for details on kinematic modeling) exhibit very nearly linear behavior because the small angle approximation holds. In fact, the small angle approximation is good to 1% up to over 7 degrees of angular misalignment. Misalignments of that magnitude should be rare in most mechanical assemblies with the exception of very flexible structures.

The linearity assumption proved excellent in the electronics assembly case study presented here. For example, Figure 3.8.2 shows the response of the number one foot position error to the angular misalignment of the lead frame. The response was estimated using a simulation of the lead forming process as documented in Appendix B. The range plotted corresponds to six standard deviations of the noise factor $\mathbf{n}_{370}$. There is no visible departure from nonlinear

behavior in this range. In fact, the response is linear to less than one part per billion within the $\pm 6\sigma$ range of the noise factor.

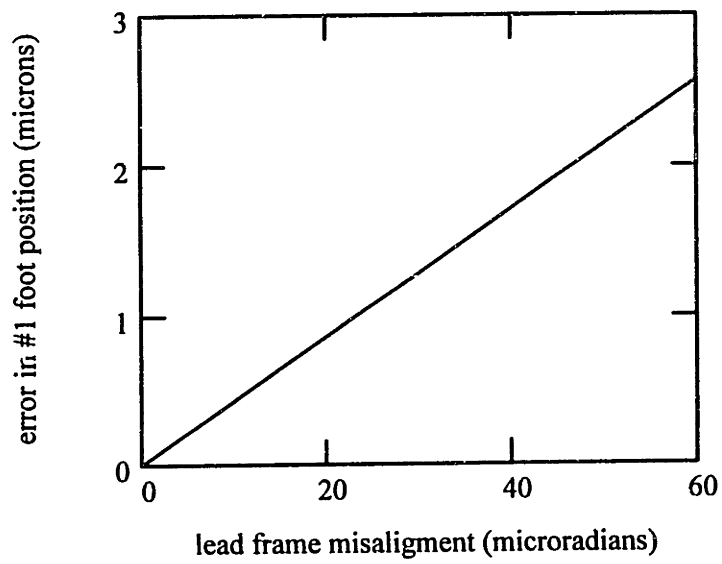


Figure 3.8.2 Linearity of response of #1 foot position error to lead frame misalignment.

Despite the excellent agreement in the MCM case study, linearity should be assumed only when dictated by good engineering judgment. The case study on dual head valve grinding in Chapter 10 will show that some noise factors extend over a wide enough range to reveal significant non-linearity in system response. The case study on CNC crankpin grinding in Chapter 11 will show that the linearity assumption does not hold for certain classes of tolerances.

3.9 Definition of the Process Capability Matrix

Much of the value of the process capability index C_p is that it captures information about tolerance and variation in a single dimensionless parameter. It is useful to maintain this dimensionless form, but noise factors and quality characteristics will often have different dimensions. This motivates the definition of the *process capability matrix*, **C**. The elements of the matrix, $\mathbf{C}_{ij}$, are defined as

$$\mathbf{C}_{ij} \equiv \frac{3\sigma_j \cdot \left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{ij}}{(U_i - L_i)/2} \tag{3.9.1}$$

where σ_j is the standard deviation of the jth noise factor and U_i and L_i are the upper and lower tolerance limits on the ith quality characteristic. It is important to note that the partial derivative in the numerator must be evaluated at the target point **t**. Different target points may yield different process capability matrices even in manufacture of the same product with the same tolerances in the presence of the same noises.

Comparison of the definitions of the capability matrix elements (Equation 3.9.1) with the capability index (Equation 1.3.1.1) reveals that the elements of the process capability matrix are the *inverse* of the capability indices for each quality characteristic with respect to each noise factor. It is therefore important to recall that with **C**, smaller is better, whereas with C_p, larger is better. This convention may seem confusing, but has many significant advantages. Many concepts from mathematics can be applied to **C** to gain insight into practical problems (as will be shown in subsequent chapters). For example:

- The Euclidean norms of the column vectors enter in formulae for predicting rolled throughput yield
- The Euclidean norms of the row vectors determine the most promising tolerances to widen
- The generalized inverse of **C** is a mapping from measured errors to on-line adjustment settings that minimize quality loss as defined by Taguchi

Another important difference between **C** and C_p is that, by definition, C_p is always positive while the elements of **C** may be negative since the partial derivative in the numerator has a sign. This is essential to the function of **C** in capturing correlation among quality characteristics. This concept will be illustrated by the case study of the electronic assembly process.

3.10 Definition of the Bias Vector

The value of the performance index C_{pk} is that it captures information about bias and variation in a single dimensionless parameter. For the purpose of multi-criteria analysis, it will prove to be

more convenient to keep information about bias and variation separate. While variation is captured in the process capability matrix **C**, bias is captured by a *bias vector* **k** defined as

$$\mathbf{k}_i \equiv \frac{2}{U_i - L_i} \sum_{j=1}^{n} \left[\left(\boldsymbol{\mu}_j - \mathbf{t}_j \right) \cdot \left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{ij} \right] \tag{3.10.1}$$

where $\boldsymbol{\mu}_j$ and $\mathbf{t}_j$ are the mean and target values of the jth noise factor and U_i and L_i are the upper and lower tolerance limits on the ith quality characteristic.

3.11 An Interpretation of the Capability Matrix

The special form of the process capability matrix and bias vector serves the purpose of simplifying analysis of multi-input multi-output systems. The convenience of **C** is best understood through a mathematical interpretation. Let us define a *normalized quality vector* $\overline{\delta\mathbf{q}}$

$$\overline{\delta\mathbf{q}}_i \equiv \frac{\mathbf{q}_i - \dfrac{U_i + L_i}{2}}{(U_i - L_i)/2} \tag{3.11.1}$$

An interesting property of the normalized quality vector is its relationship to average quality loss as defined by Taguchi. Section 8.3 demonstrates that the average quality loss is proportional to the square of the Euclidean norm of $\overline{\delta\mathbf{q}}$.

By definition in Equation 3.11.1, the normalized quality vector conforms to its tolerance limits only if all of its elements are between -1 and 1. Each instance of the normalized quality vector $\overline{\delta\mathbf{q}}$ is associated with a discrete unit of product. That discrete unit of product conforms to the acceptance criteria if and only if $\overline{\delta\mathbf{q}}$ lies within a hypercube as depicted for the 2D case in Figure 3.11.1. Let us call this hypercube the *tolerance range TR*.

Definition The *tolerance range* is the set

$$TR = \left\{ \overline{\delta\mathbf{q}} \;\middle|\; \left\| \overline{\delta\mathbf{q}} \right\|_\infty \geq 1 \right\} \tag{3.11.2}$$

Given this definition of the tolerance range *TR*, the event of conformance is $\overline{\delta\mathbf{q}} \in TR$.

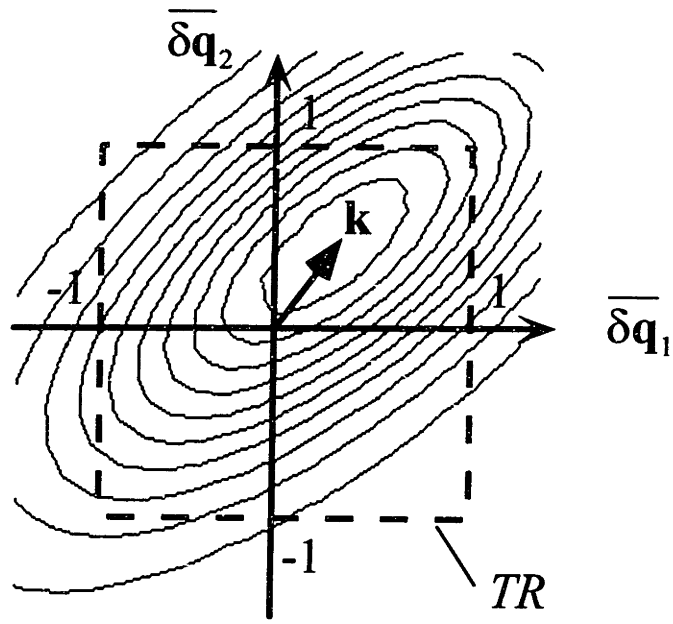


Figure 3.11.1 The space of normalized quality characteristics.

Let us also define a *normalized noise vector* $\overline{\delta \mathbf{n}}$

$$\overline{\delta \mathbf{n}}_j \equiv \frac{\mathbf{n}_j - \mu_j}{3\sigma_j}. \tag{3.11.3}$$

Therefore $\overline{\delta \mathbf{n}}$ is a vector of random variables each of which has a mean of zero and a standard deviation of 1/3.

$$E\left(\overline{\delta \mathbf{n}}_j\right) = 0 \text{ for all } j \in 1 \ldots n \tag{3.11.4}$$

$$\sqrt{E\left(\left[\overline{\delta \mathbf{n}}_j - E\left(\overline{\delta \mathbf{n}}_j\right)\right]^2\right)} = \frac{1}{3} \text{ for all } j \in 1 \ldots n. \tag{3.11.5}$$

The space of normalized noise factors is depicted graphically in Figure 3.11.2. If the noise factors are all normally distributed and probabilistically independent, then all the surfaces of constant probability density will be hyper-spheres (or circles in 2D as shown in Figure 3.11.2).

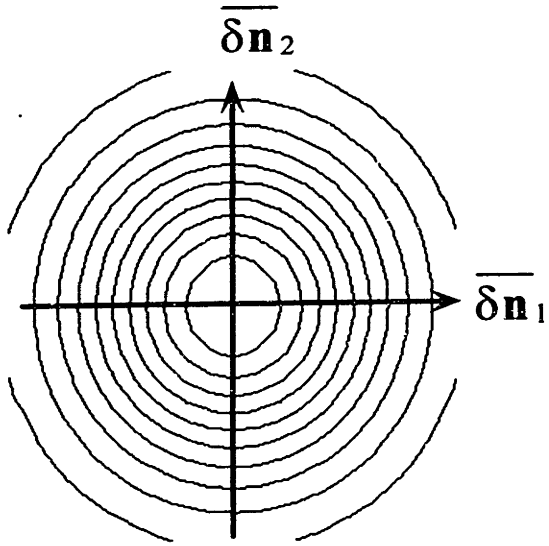


Figure 3.11.2 The space of normalized noise factors.

With the definitions of normalized quality (Equation 3.11.1) and normalized noise (Equation 3.11.2), the linearized representation of the manufacturing system in Equation 3.8.1 becomes

$$\overline{\delta \mathbf{q}} = \mathbf{C} \cdot \overline{\delta \mathbf{n}} + \mathbf{k} \ . \tag{3.11.6}$$

Therefore the capability matrix **C** and bias vector **k** can be viewed as an affine mapping from a vector of normalized noise into a vector of normalized quality.

The affine transformation of the noise vector creates a space of m random variables. This normalized quality space is depicted in Figure 3.11.2. If the noise space is Gaussian as depicted in Figure 3.11.2, then the spheres of constant probability density in $\overline{\delta \mathbf{n}}$ space will transform into ellipsoids in $\overline{\delta \mathbf{q}}$ space.

The interpretation of the capability matrix and bias vector as described in this section leads to a simple interpretation of the vector **k**. The expected value of a sum of random variables is the sum of the expected values of the random variables. From this fact and Equations 3.11.3 and 3.11.5 one may deduce that the expected value of the quality characteristic vector is the bias vector

$$E\left(\overline{\delta \mathbf{q}}\right) = \mathbf{k} \, . \tag{3.11.7}$$

It will prove to be useful to define which points in $\overline{\delta\mathbf{n}}$ space map into conforming discrete units of product. This leads to the definition of the *tolerance domain TD*.

Definition The *tolerance domain TD* is the pre-image of the tolerance range *TR* under the affine mapping $\overline{\delta\mathbf{q}} = \mathbf{C} \cdot \overline{\delta\mathbf{n}} + \mathbf{k}$.

$$TD = \left\{ \overline{\delta\mathbf{n}} \;\middle|\; \left\| \mathbf{C} \cdot \overline{\delta\mathbf{n}} + \mathbf{k} \right\|_{\infty} \leq 1 \right\} \tag{3.11.8}$$

Given this definition of the tolerance domain *TD*, the event of conformance of a discrete unit of product is $\overline{\delta\mathbf{n}} \in TD$. This definition will be a key to computing probability of conformance in Chapter 4.

3.12 Steps for Constructing the Process Capability Matrix and Bias Vector

There are many ways to go about constructing the process capability matrix (**C**) and bias vector (**k**) for a manufacturing system. **C** and **k** are fully defined by Equations 3.9.1 and 3.10.1 and any method consistent with those definitions will allow one to use the analytical techniques presented in the rest of this thesis. However, for clarity of exposition, the following steps are suggested:

1) Identify the quality characteristics used in the product's acceptance criteria.
2) Identify a set of noise factors that are likely to cause variation in the quality characteristics identified in step 1. Estimate the form of the probability density functions over the noise factors and their standard deviations.
3) Estimate the sensitivity of each quality characteristic to small changes in each noise factor. This may be accomplished more efficiently using designed experiments. Often the sensitivity can be determined analytically or with computer models of the process.
4) Assemble **C** and **k** as defined in Equations 3.9.1 and 3.10.1 above.

3.13 The Capability Matrix for Surface Mount of MCMs

This section reviews the construction of the capability matrix for the surface mount of MCMs. Steps 1 and 2 in the procedure outlined in section 3.11 were completed in sections 3.3 and 3.6 respectively. Therefore, the next step is to estimate the sensitivities of quality characteristics

with respect to noise factors. To make the process more manageable, I will break the surface mount process into three separate processes: lead frame manufacture, component assembly / lead forming, and placement of the MCM onto the PWA.

Each of the processes is affected by different sets of noise factors. Therefore, the capability matrices developed in the next three sections are submatrices within the larger capability matrix for the complete set of noise factors. The procedure by which they can be assembled to create the complete capability matrix for the system will be described in Chapter 4.

3.13.1 Lead Frame Manufacture

This section will develop a process capability matrix and bias vector for the process by which the lead frame was manufactured. The noise factors that affect this process have already been identified in Section 3.6. The factors relevant to this process are numbered 1 through 369.

Due to variations in the stamping process, there will be some variation in each lead position in the unformed lead frame. As discussed in section 3.6, this variation is represented by noise factors $\mathbf{n}_1$ through $\mathbf{n}_{368}$. Due to the way that these noise factors were defined, estimating sensitivity is trivial. The Jacobian of the quality characteristics with respect to the noise factors will be the identity matrix. This reflects the fact that any variations downstream tend to linearly superpose onto these variations rather than amplify them or otherwise interact with them to any significant degree.

Recall from Section 2.4 that there tends to be some variation and bias in overall scale of the lead frame. A lead frame that is too large tends to cause a linear increase in side-to-side error on each side of the MCM as depicted in Figure 3.12.1.1. I describe the variation in scale of the lead frame with the random variable $\mathbf{n}_{369}$ and list its mean and standard deviation in Table 3.6.1.

The sensitivity of the quality characteristics with respect to scaling can be computed by a simple transformation. If one assigns a Cartesian coordinate system to the center of the electronics package, then the foot positions will each have some nominal coordinates. One may transform the positions of the feet by multiplying by a scaling factor. The resulting error in the side-to-side positions of the feet will yield the desired sensitivities.

Applying the sensitivities described above in concert with the definition of the capability matrix and bias vector we find

$$\mathbf{C} = \left[0.35 \cdot \mathbf{I}_{368 X 368} \middle| 3.85 \times 10^{-4} \begin{Bmatrix} (i-63) \cdot \mathbf{1}_{125} \\ (i-155) \cdot \mathbf{1}_{59} \\ (i-247) \cdot \mathbf{1}_{125} \\ (i-339) \cdot \mathbf{1}_{59} \end{Bmatrix} \right] \tag{3.13.1.1}$$

$$\mathbf{k} = 3.85 \times 10^{-3} \begin{Bmatrix} (i-63) \cdot \mathbf{1}_{125} \\ (i-155) \cdot \mathbf{1}_{59} \\ (i-247) \cdot \mathbf{1}_{125} \\ (i-339) \cdot \mathbf{1}_{59} \end{Bmatrix} \tag{3.13.1.2}$$

where $\mathbf{1}_{125}$ denotes a vector of 125 ones and i denotes the row number of that element within the matrix. The presence of i in the matrix captures the linear dependence of side to side error on lead number caused by scaling of the lead frame.

3.13.2 Component Assembly and Lead Forming

The lead frame is assembled onto the ceramic body of the MCM and brazed in place. There may exist some error in the x and y position and in the orientation of the lead frame with respect to the body. These three noise factors are denoted as $\mathbf{n}_{371}$, $\mathbf{n}_{372}$, and $\mathbf{n}_{373}$ respectively and are listed in Table 3.6.1.

As discussed in Section 2.4, the misalignment of the lead frame will cause the shins of the ribbon leads to be vertically misaligned. This vertical misalignment, in turn, adds to the error in side to side position of the feet. Therefore, some care is required in determining the sensitivity of the final foot positions to the misalignment of the lead frame. We employed a simulation of the lead forming process to compute the sensitivities (the simulation is discussed in detail in Appendix B). Based on these and on the means and standard deviations listed in Table 3.6.1, one may construct a process capability matrix and bias vector for the lead frame assembly and lead forming process.

$$\mathbf{C} = \left[\begin{array}{c|c|c} 0.838 \cdot \mathbf{1}_{59} & 0.20 \cdot \mathbf{1}_{59} & \mathbf{0}_{59} \\ \hline 1.529 \cdot \mathbf{1}_{125} & \mathbf{0}_{125} & 0.20 \cdot \mathbf{1}_{125} \\ \hline 0.838 \cdot \mathbf{1}_{59} & -0.20 \cdot \mathbf{1}_{59} & \mathbf{0}_{59} \\ \hline 1.529 \cdot \mathbf{1}_{125} & \mathbf{0}_{125} & -0.20 \cdot \mathbf{1}_{125} \end{array} \right] \tag{3.13.2.1}$$

$$\mathbf{k} = \begin{Bmatrix} 0.10 \cdot \mathbf{1}_{59} \\ \hline 0.10 \cdot \mathbf{1}_{59} \\ \hline -0.10 \cdot \mathbf{1}_{59} \\ \hline -0.10 \cdot \mathbf{1}_{59} \end{Bmatrix} \tag{3.13.2.2}$$

3.13.3 Placement of the MCM onto the PWA

After the ribbon leads are formed into the appropriate shape, the MCM is moved to an assembly robot. The robot grasps the MCM by the edges of the ceramic body and places it on a printed wiring assembly. The robot itself has limited accuracy and repeatability due to thermal errors, resolution of its angular encoders, and a host of other effects. Additional error is introduced by the contact between the MCM body and the robotic gripper. The resulting errors in x and y position and in orientation of the MCM are denoted as $\mathbf{n}_{374}$, $\mathbf{n}_{375}$, and $\mathbf{n}_{376}$ respectively and are listed in Table 3.6.1. The sensitivities of the side-to-side errors in the lead positions can be calculated based on rigid body transformations. The sensitivities and the noise data in Table 3.6.1 can be used to compute the process capability matrix and bias vector for the robotic assembly procedure.

$$\mathbf{C} = \left[\begin{array}{c|c|c} 0.231 \cdot \mathbf{1}_{59} & 0.10 \cdot \mathbf{1}_{59} & \mathbf{0}_{59} \\ \hline 0.462 \cdot \mathbf{1}_{125} & \mathbf{0}_{125} & 0.10 \cdot \mathbf{1}_{125} \\ \hline 0.231 \cdot \mathbf{1}_{59} & -0.10 \cdot \mathbf{1}_{59} & \mathbf{0}_{59} \\ \hline 0.462 \cdot \mathbf{1}_{125} & \mathbf{0}_{125} & -0.10 \cdot \mathbf{1}_{125} \end{array}\right] \tag{3.13.3.1}$$

$$\mathbf{k} = \mathbf{0}_{368} \tag{3.13.3.2}$$

3.14 Reading Process Capability Matrices

The capability matrices above provide an opportunity to discuss the physical interpretation of these mathematical objects.

The columns of the matrices correspond to noise factors. Each column indicates the change in the normalized quality characteristics given that the noise factor is set to three standard deviations from its target value and all the other noise factors are at their target values. For

example, consider the first column of the capability matrix for lead forming (Equation 3.13.2.1). It corresponds to noise factor $\mathbf{n}_{371}$, the misalignment of the lead frame on the component body. The first submatrix displayed in this column relates to the #1 side of the MCM, the second element relates to the #2 side, et cetera. The elements that are greater than one indicate a problem. When the misalignment of the lead frame on the component body is at its 3σ value of 30μrad (see Table 3.6.1), the #2 and #4 sides of the MCM will not conform to their tolerances. The #1 and #3 sides of the MCM will be barely within their tolerance bands.

The rows in the matrices correspond to quality characteristics. In this case, each row indicates the sensitivity of the error in side to side locations of the foot whose number corresponds to the row number of the matrix. For example, consider the capability matrix for lead placement (Equation 3.13.3.1). The leads on the #1 side correspond to the first row of the matrix. As one would expect, they are affected by x positioning accuracy of the robot, but not by the y positioning accuracy.

The values of the individual elements of the capability matrices have clear implications for the performance of the manufacturing system. The facts listed below follow from the developments in Chapter 4, but are previewed here to give the reader a feel for the capability matrix:

- A value of 2.0 in any element indicates a serious quality problem. The rolled throughput yield cannot be higher that 86.6% if any element of $\mathbf{C}$ exceeds 2.0 (assuming Gaussian noise).
- A value of 1.0 corresponds to a reasonable level of process capability. If the only large element of the matrix is a 1.0, then the yields tend to be about 99.7%.
- A value of 0.5 corresponds to excellent process capability. If all the elements of the matrix are 0.5 or smaller, then the process should approach 6σ quality and defect rates should be on the order of $2n$ parts per billion (where n is the number of noise factors).

The elements of the bias vector are the expected values of the normalized quality characteristics. They sum the effects of all the noise factors. The facts listed below follow from the developments in Chapter 4, but are previewed here to give the reader a feel for the bias vector:

- A bias vector value of 1.0 is an indication of a serious deficiency. The rolled throughput yield cannot be better that 50% if any element of **k** exceeds 1.0.
- If a process is in statistical control as defined by Shewhart (1931), then the mean of the process will be within $\pm 3\sigma$ of the target values for all of the quality characteristics. Therefore, in a process under statistical control, every element of **k** will be less than the Euclidean norm of the corresponding row of **C**.

3.15 Chapter Conclusions

This chapter has introduced the concepts of the capability matrix and bias vector – dimensionless representations of a manufacturing system's response to noise factors. The concepts of quality characteristics and noise factors, established in the field of robust design, were adapted to suit the purposes of manufacturing system design. The quality characteristics and noise vectors were assembled into vectors and manufacturing systems were modeled as vector transformations. The capability matrix and bias vector were defined and interpreted in light of this view of manufacture as a vector transformation. Capability matrices were developed for the processing steps that comprise the manufacturing system for surface mount of MCMs. The next chapter introduces manufacturing block diagrams. These block diagrams will allow the capability matrices and bias vectors for the individual processing steps to be assembled into a single capability matrix and bias vector for the entire manufacturing system.

4. Manufacturing System Block Diagrams

This section introduces the use of process capability matrices and bias vectors in manufacturing system block diagrams. One may model a manufacturing system by generating capability indices and bias vectors for each processing step. Each matrix and vector is represented in a block. Connections among the blocks represent the propagation of dimensional variation within the manufacturing system.

Most products are manufactured by processes comprised of many steps each of which can impact final product quality. The variations introduced in one step may be added to, amplified, or reduced by subsequent processing steps. These interactions can have a significant impact on the performance of manufacturing systems as measured by yield, variance, and bias. This chapter addresses this need by developing a new technique for modeling manufacturing systems using block diagrams. Chapter 8 will develop methods to represent on-line adjustment procedures in manufacturing block diagrams.

4.1 Review of MIMO State Space Block Diagrams

A block diagram of a system is a pictorial representation of the functions performed by each component and the flow of signals among the components. Compared to a purely mathematical representation, a block diagram has the advantage of indicating more realistically the signal flows in the physical system [Ogata, 1990]. A block diagram can retain the full rigor of abstract mathematical representations if there exist unambiguous isomorphisms between the block diagrams and the mathematical representations. This is the case with state space block diagrams of dynamic systems.

State space block diagrams of multi-input multi-output (MIMO) dynamic systems are useful in design and analysis of complex systems. Individual blocks represent transformations among input vectors, state vectors, and output vectors. Any system governed by a system of linear differential equations can be represented in this formalism. For example, the first order linear system

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{4.1.1}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \tag{4.1.2}$$

can be represented by the block diagram in Figure 4.1.1. The thick lines in the figure indicate that the signals propagating through the system are vectors.

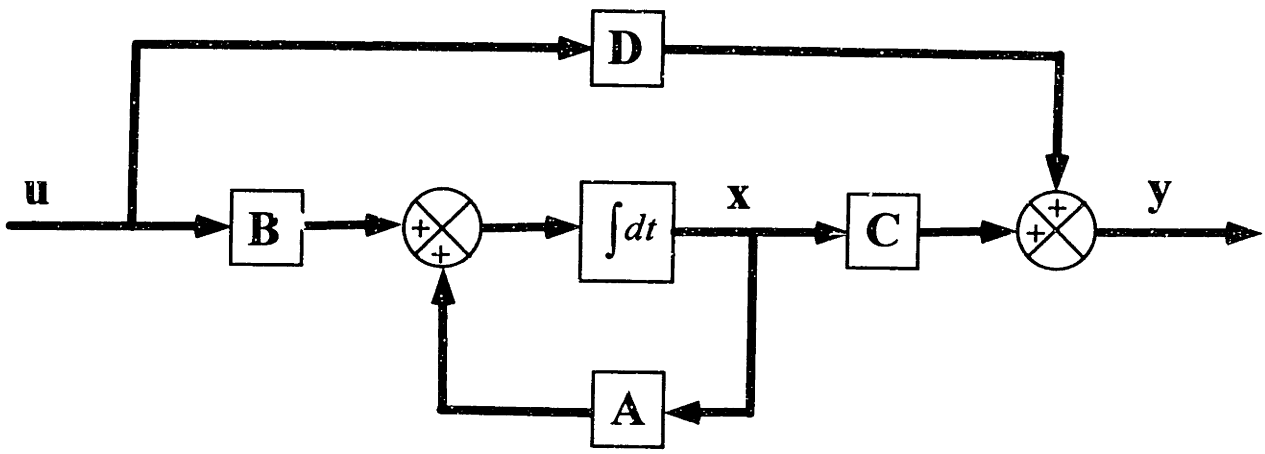


Figure 4.1.1 A state space block diagram of a first order LTI MIMO system.

Figure 4.1.1 and Equations 4.1.1 and 4.1.2 represent a canonical form for a linear time invariant (LTI) multi-input-multi-output (MIMO) system. The canonical form is convenient in that a rich body of theory exists concerning the salient behaviors of such systems (e.g. their stability, controllability, and observability). To apply this body of theory to a given LTI MIMO system, it is convenient to reduce the system to the canonical form.

For MIMO state space diagrams, there exist a set of reduction rules derived from linear algebra. For example, the system of two parallel linear transformations may be reduced to an equivalent system as shown in Figure 4.1.2. By repeated application of such reduction rules, any first order LTI MIMO system can be reduced to the canonical form of Figure 4.1.1. In the next section we define a canonical form and reduction rules for manufacturing systems analysis.

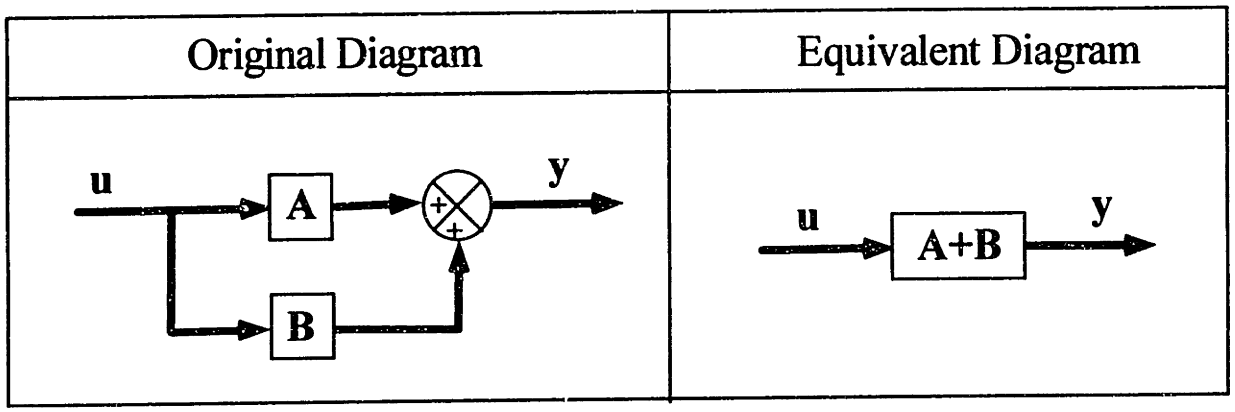


Figure 4.1.2 Reduction rule for a parallel connection of blocks.

4.2 A Canonical Form for a Manufacturing System

The advantages of state space block diagrams is substantial. Block diagrams allow intuitively clear modeling of physical systems. The reduction rules permit representation in canonical form. The canonical form, in turn, provides a standard system for which a body of theory can be developed. To develop a similar body of theory for manufacture, we seek to adapt state space block diagramming for use in analyzing yield, quality, and dimensional variation. The concept of the process capability matrix can form a basis for this project.

Recall from section 3.11 that the process capability matrix can be interpreted as an affine transformation from a normalized noise space into a normalized quality characteristic space (Equation 3.11.5). Using the formalism of MIMO state space block diagrams, Equation 3.11.5 is equivalent to the diagram in Figure 4.2.1. This is the canonical form to which more complicated system diagrams must be reduced.

The system depicted in Figure 4.2.1 is a memoryless LTI MIMO system; It has no dynamic behavior. The interesting aspects of this system stem from the nature of its input and output signals. As discussed in Section 3.11, one of the input signals, $\overline{\delta \mathbf{n}}$, is a vector of independent normalized discrete random variables. The elements take on randomly selected values for each trial (each trial represents the manufacture of a discrete unit of product). The input signal $\overline{\delta \mathbf{n}}$ generates the random component of error in the output signal. The other input signal is simply the number '1'. It generates the bias in the output signal. The output signal is normalized so that

success on each trail is determined by the infinity norm of the output vector as discussed in Section 3.11. Chapter 5 explores the dependence of rolled throughput yield on **C** and **k** of any system representable in the canonical form of Figure 4.2.1.

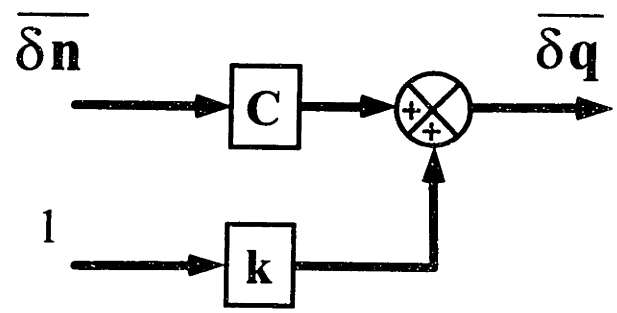


Figure 4.2.1 The canonical representation of a manufacturing system.

4.3 Parallel Connection of Manufacturing Processes

It is often the case that two processing steps will have an effect on the same set of quality characteristics, but are affected by different sets of noise factors. A system of two processes with this relationship is representable as a parallel connection as depicted in Figure 4.3.1. Here we employ a subscript notation in which $\overline{\delta \mathbf{n}}_{x...y}$ denotes a vector composed of the elements $\overline{\delta \mathbf{n}}$ with indices x through y inclusive. Stated differently, the indices denote the relationship of the columns of the capability matrices to a table of noise factors (e.g. Table 3.6.1).

The block diagram in the right hand plane of Figure 4.3.1 is a mathematically equivalent reduction of the block diagram in the left hand plane. This can be confirmed by a trivial application of linear algebra. The reduction rule will prove useful in analyzing the surface mount of MCMs in the next section and again in Chapter 8.

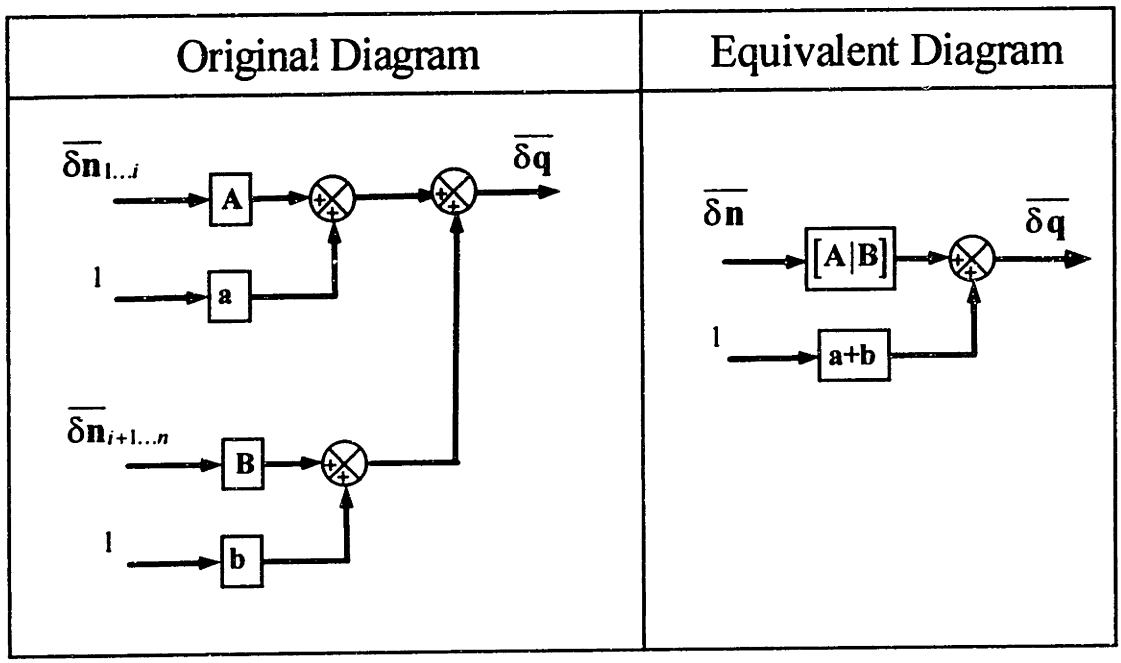


Figure 4.3.1 Parallel connect of manufacturing processes.

4.4 Reduction of the MCM Process

To provide an example of the reduction rule for parallel processes, consider the surface mount process described in Chapter 2. This system is composed of three processes each characterized by a process capability matrix and bias vector as developed in Section 3.13. The entire manufacturing system can be viewed as a parallel connection of these three steps as depicted in Figure 4.4.1. This system can be reduced to the canonical form of Figure 4.2.1 which provides the process capability matrix and bias vector of the entire system.

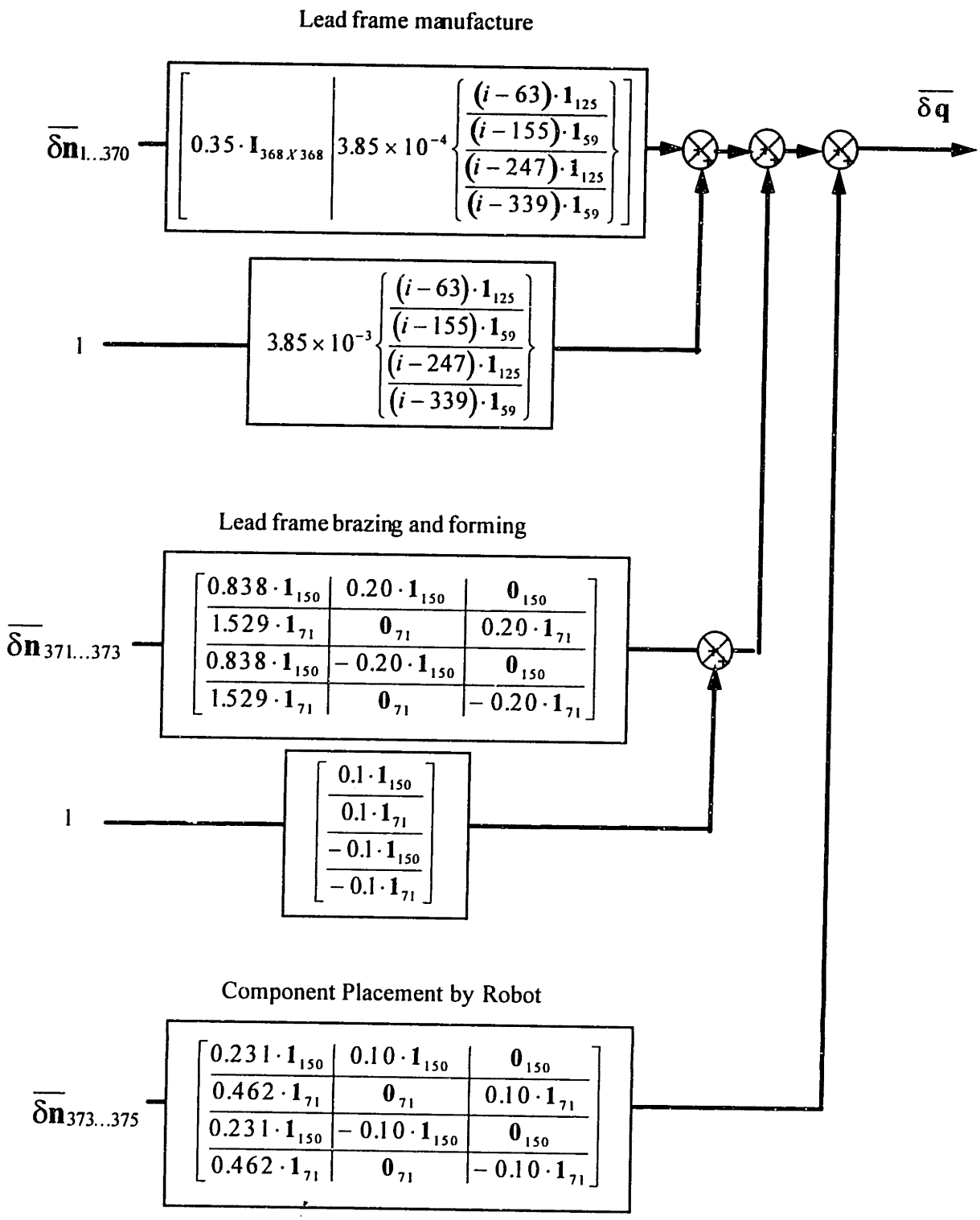


Figure 4.4.1 Block diagram of the electronic assembly process.

The reduction rule of Figure 4.3.1 was applied twice to the block diagram of Figure 4.4.1. The resulting capability matrix of the system is a bit unwieldy so it is not presented here explicitly. Instead, a slightly reduced equivalent matrix is given using a reduction rule developed in Section 5.6 (Lemma 5.6.1). If the cross product of two columns of **C** is zero and the noises are independent and normally distributed, then those two columns of **C** may be combined into one column composed of the square root of the sum of the squares of the elements in each row of the original two columns times the sign of the element in the first column. Employing this rule to combine column 372 with 374 and column 373 with 375 yields an equivalent capability matrix for the system in Figure 4.4.1.

$$\mathbf{C} = \left[0.35 \cdot \mathbf{I}_{368 X 368} \middle| 3.85 \times 10^{-4} \left\{ \begin{array}{c} (i-63)\cdot \mathbf{1}_{125} \\ \hline (i-155)\cdot \mathbf{1}_{59} \\ \hline (i-247)\cdot \mathbf{1}_{125} \\ \hline (i-339)\cdot \mathbf{1}_{59} \end{array} \right\} \middle| \begin{array}{c|c|c|c} 0.838\cdot \mathbf{1}_{125} & 0.231\cdot \mathbf{1}_{125} & 0.22\cdot \mathbf{1}_{125} & \mathbf{0}_{125} \\ \hline 1.529\cdot \mathbf{1}_{59} & 0.462\cdot \mathbf{1}_{59} & \mathbf{0}_{59} & 0.22\cdot \mathbf{1}_{59} \\ \hline 0.838\cdot \mathbf{1}_{125} & 0.231\cdot \mathbf{1}_{125} & -0.22\cdot \mathbf{1}_{125} & \mathbf{0}_{125} \\ \hline 1.529\cdot \mathbf{1}_{59} & 0.462\cdot \mathbf{1}_{59} & \mathbf{0}_{59} & -0.22\cdot \mathbf{1}_{59} \end{array} \right] \quad (4.4.1)$$

The system bias vector is

$$\mathbf{k} = 3.85 \times 10^{-3} \left\{ \begin{array}{c} (i-63)\cdot \mathbf{1}_{125} \\ \hline (i-155)\cdot \mathbf{1}_{59} \\ \hline (i-247)\cdot \mathbf{1}_{125} \\ \hline (i-339)\cdot \mathbf{1}_{59} \end{array} \right\} + \left\{ \begin{array}{c} 0.1\cdot \mathbf{1}_{125} \\ \hline 0.1\cdot \mathbf{1}_{59} \\ \hline -0.1\cdot \mathbf{1}_{125} \\ \hline -0.1\cdot \mathbf{1}_{59} \end{array} \right\}. \quad (4.4.2)$$

The capability matrix and bias vector displayed above are a parsimonious model of the MCM surface mount process. The relationship between the model and the physical system is explored in the following section.

4.5 Comparison of the Model to Experimental Data

The capability matrix and bias vector in Equations 4.4.1 and 4.4.2 when combined with the affine transformation of Equation 3.9.1 ($\overline{\delta \mathbf{q}} = \mathbf{C} \cdot \overline{\delta \mathbf{n}} + \mathbf{k}$) comprise a model of the probabilistic behavior of dimensional variation in the surface mount process. A very simple Monte Carlo simulation of the process can be performed by using pseudo-random number generators to create instances of the normalized noise vector. This can be accomplished by generating vectors of independent random variables from distributions with standard deviations of one third and means

of zero. These vectors are then transformed using the capability matrix and bias vector. The result is a normalized quality vector for each statistical trial.

Such a Monte Carlo simulation was performed with one hundred sets of subgroup sizes of nine. A histogram of the results from one of these subgroups is plotted in Figure 4.5.1. The normalized quality characteristics of the leads were grouped together without regard to lead number. The abscissa is the normalized quality characteristic and is dimensionless. Bins of width 0.02 were defined over the abscissa. The ordinate is the frequency – the number of measurements that fell within a bin nearest the given quality characteristic value. A similar plot was made using the experimental data and is displayed in Figure 4.5.2. Comparing the two figures, the qualitative impression is that the data and the model have a similar variance and distribution.

The subjective impression that the data is similar to the data in mean and variance is borne out by computation. By a student's-t test, the hypothesis that the mean of the data was selected from the population of the means of the Monte Carlo subgroups could not be rejected at α=0.01. Similarly, by a χ^2 test, the hypothesis that the variance observed in the data was the same as that observed in the Monte Carlo subgroups could not be rejected at α=0.01. This suggests that the mean and variance in the data is consistent with the mean and variance of the model if one ignores the correlation among the leads.

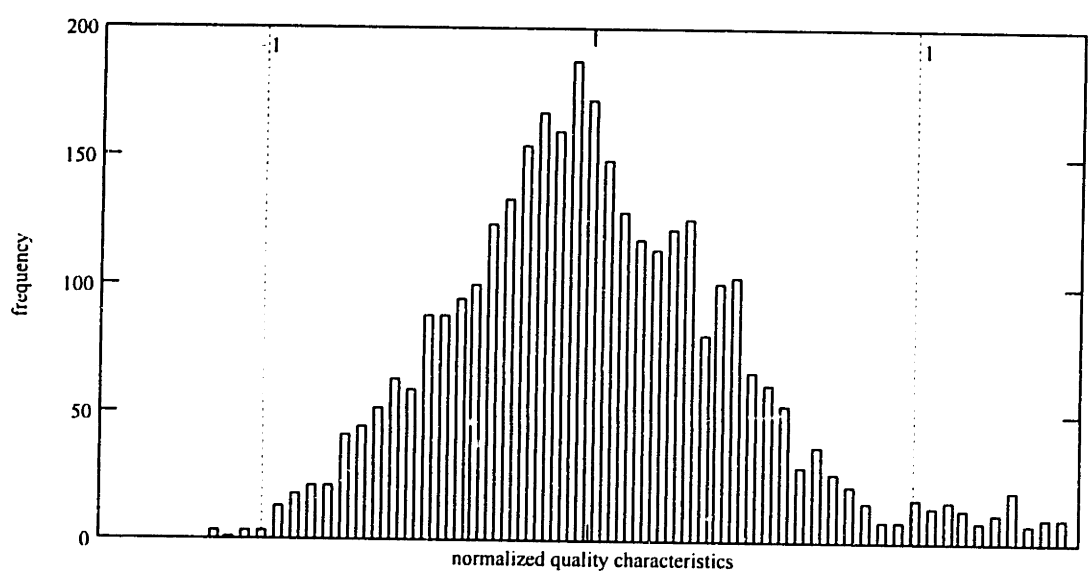


Figure 4.5.1 Simulation results for lead position error from placement of nine 368 leaded MCMs – no adjustments applied.

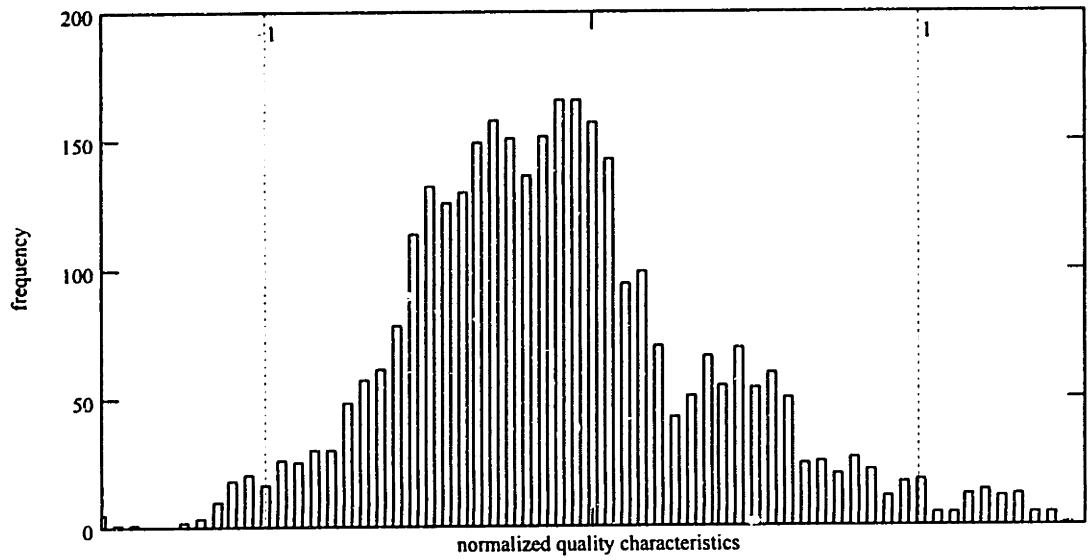


Figure 4.5.2 Data on lead position error from placement of nine 368 leaded MCMs – no adjustments applied.

The fact that the mean and variance of the model match that of the data is important. It ensures that the model will correctly predict the traditional measures of system performance such as process capability C_p, bias factor k, and first time yield Y_{FT}. However, the fact that the means and variances of the model fit that of the data is not surprising. A much simpler model of the data can be made to have this property. For example, the traditional six sigma producibility analysis model assumes all the quality characteristics are probabilistically independent. If one assumes that they are normally distributed, then a statistical model with two parameters (mean and variance) can obviously be fit to the mean and variance in the data. What makes the capability matrix model compelling is that it also does a reasonable job of capturing the correlation among the quality characteristics. This can be demonstrated by studying patterns in the data from individual MCMs.

The result of a single trial of a Monte Carlo simulation is presented in Figure 4.5.3. The side to side variation in the #1 side of the simulated discrete unit of product is displayed versus lead number. This may be compared to a similar plot of measurements taken from an actual MCM (Figure 4.5.4). Using the plots, one may make a qualitative assessment of the match between the behavior of the model and that of the actual experimental system.

In some ways, the match between the model behavior and experimental data is excellent. For example, one may observe that the linear trend in the data has been captured fairly well. On the other hand, there seems to be a significantly stronger correlation between neighboring leads in the experimental data than in the model. For example, the correlation coefficient between the #2 and #3 leads in the data is 0.954 while in the model it is only 0.868.

The key point at this stage is that the capability matrix for the surface mount process is a fairly simple, physically based model of the manufacturing system performance that captures most of the salient statistical features of the output. The statistical features of the output that this capability matrix fails to capture will prove to be unimportant in system design. This will be demonstrated in Chapters 5, 6, and 7.

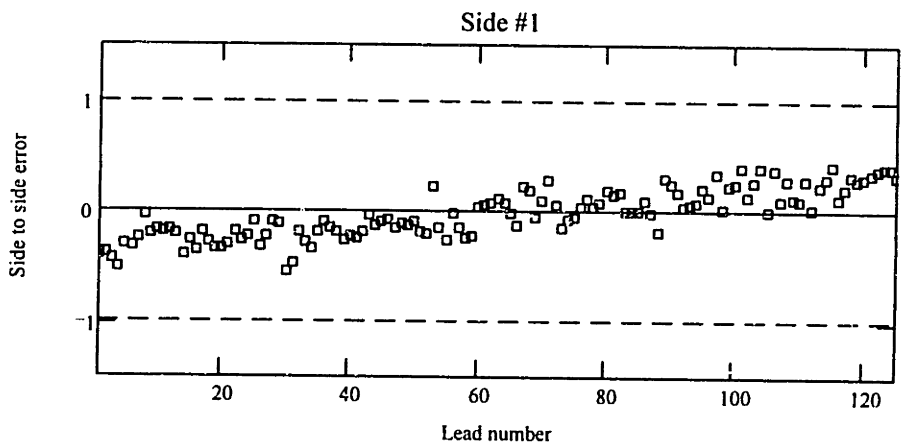


Figure 4.5.3 Result from a single trial of a Monte Carlo simulation, side #1.

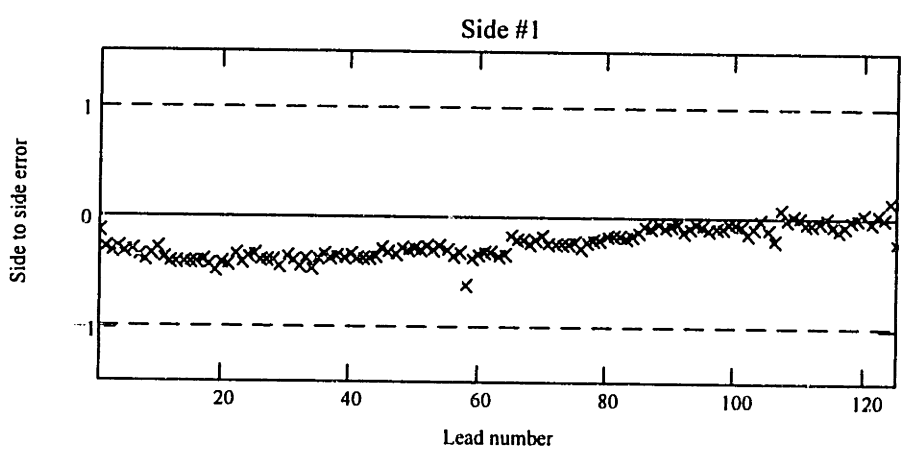


Figure 4.5.4 Result from a single trial of an actual experiment, side #1.

4.6 Chapter Conclusions

This chapter has introduced the concept of manufacturing system block diagrams – graphical representations of the flow of variation in a manufacturing system. The block diagrams are adaptations of MIMO block diagrams from control theory. They obey the same reduction rules, however, each block contains a capability matrix or bias vector. The input signals to capability matrix blocks are always normalized noise vectors. The inputs to bias vector blocks are always unity.

This chapter has shown how to represent a system comprised of several processes as a parallel connection of multiple systems. It also showed how to reduce such a representation to a single capability matrix / bias vector pair. This procedure was applied to the surface mount of MCMs. The capability matrices and bias vectors developed in Chapter 3 for three different processes were reduced to a single **C k** pair.

The next chapter introduces algorithms and equations for mapping these **C k** pairs into rolled throughput yield. This is essential for making system design decisions since rolled throughput yield is a key system performance metric. System block diagrams will be employed again in Chapter 9 to demonstrate the value of adjustment procedures in improving rolled throughput yield.

5. Rolled Throughput Yield

In this chapter, I explore a key measure of the ability of a manufacturing system to meet product tolerances – rolled throughput yield. I will define the measure mathematically, and then discuss means for computing its value based on the process capability matrix **C** and bias vector **k**. I will develop algorithms that are very general as well as estimates and bounds for a variety of special cases.

5.1 Rolled Throughput Yield Defined

It is not conformance to individual acceptance criteria that determines product acceptance. Engineering specifications generally require *simultaneous* conformance to *all* criteria. Therefore, an appropriate measure of manufacturing system performance is *rolled throughput yield* (Y_{RT}) [Harry and Lawson, 1992]. We may now define *rolled throughput yield* and express it in mathematical terms as a function of the quality vector **q**, and the associated tolerance limits L and U.

Definition The *rolled throughput yield* of a manufacturing system is the probability that all the quality characteristics of any discrete unit of product will meet their associated tolerance limits.

$$Y_{RT} \equiv \Pr(L_i \le \mathbf{q}_i \le U_i \text{ for all } i \in 1,\ 2,\ \ldots m). \tag{5.1.1}$$

It follows directly from the definition of the joint density function that

$$Y_{RT} = \int_{L_1}^{U_1}\int_{L_2}^{U_2}\cdots\int_{L_m}^{U_m} p_q(\mathbf{q}_1,\mathbf{q}_2,\ldots,\mathbf{q}_m)\mathrm{d}\mathbf{q}_1\mathrm{d}\mathbf{q}_2\ldots\mathrm{d}\mathbf{q}_m\ . \tag{5.1.2}$$

Equation 5.1.2 is rarely of direct use in estimation of Y_{RT}. In most cases, the joint density function over the quality characteristics $p(\mathbf{q}_1,\mathbf{q}_2,\ldots,\mathbf{q}_m)$ is not known explicitly. However, the important features of this density function may be estimated by developing a capability matrix and bias vector. The next section shows the relationship of Y_{RT} to the capability matrix model.

5.2 Relation to the Capability Matrix and Bias Vector

Section 3.4 introduced an interpretation of the capability matrix and bias vector as an affine transformation from normalized noise to normalized quality. This section will show how that result can be applied to computing rolled throughput yield.

To recapitulate the results of Section 3.11, if a complete set of noise factors can be identified and the manufacturing system's response to noise is nearly linear, then manufacture can be modeled as the affine transformation $\overline{\delta\mathbf{q}} = \mathbf{C}\cdot\overline{\delta\mathbf{n}} + \mathbf{k}$. The term $\overline{\delta\mathbf{q}}$ in this expression is a normalized quality vector defined by Equation 3.11.1. The definition of the normalized quality vector and the affine model may be combined with the definition of rolled throughput yield (Equation 5.1.1) to give

$$Y_{RT} = \Pr\left(\left\|\mathbf{C}\cdot\overline{\delta\mathbf{n}} + \mathbf{k}\right\|_\infty \le 1\right). \tag{5.2.1}$$

where

$$E\left(\overline{\delta\mathbf{n}}_j\right) = 0 \text{ for all } j \in 1\ldots n \tag{5.2.2}$$

and

$$\sqrt{E\left(\left[\overline{\delta\mathbf{n}}_j - E\left(\overline{\delta\mathbf{n}}_j\right)\right]^2\right)} = \frac{1}{3} \text{ for all } j \in 1\ldots n. \tag{5.2.3}$$

There are two ways to express rolled throughput yield as an integral. One is to integrate in normalized noise space, the other is to integrate in normalized quality space. Both will prove to have utility in different contexts.

To compute Y_{RT} in normalized noise space, recall that the event of conformance of a discrete unit of product is $\overline{\delta\mathbf{n}} \in TD$. Therefore, it follows from the definition of the probability density function that the rolled throughput yield is

$$Y_{RT} = \Pr\left(\overline{\delta\mathbf{n}} \in TD\right) = \int_{TD} p_n\left(\overline{\delta\mathbf{n}}\right) d\overline{\delta\mathbf{n}} \tag{5.2.4}$$

where $TD = \left\{\overline{\delta\mathbf{n}} \,\middle|\, \left\|\mathbf{C}\cdot\overline{\delta\mathbf{n}} + \mathbf{k}\right\|_\infty \le 1\right\}$. The advantage of this formulation is that it is generally possible to find a set of noise factors that are probabilistically independent. Therefore the joint density function is separable so that

$$Y_{RT} = \Pr\left(\overline{\delta\mathbf{n}} \in TD\right) = \int_{TD} p_n\left(\overline{\delta\mathbf{n}}_1\right) p_n\left(\overline{\delta\mathbf{n}}_2\right) \cdots p_n\left(\overline{\delta\mathbf{n}}_n\right) d\overline{\delta\mathbf{n}} \tag{5.2.5}$$

However, Equation 5.2.5 is not generally amenable to symbolic solution since the limits of integration are coupled.

The second option is to compute Y_{RT} in normalized quality space. Recall that the event of conformance of a discrete unit of product is $\overline{\delta\mathbf{q}} \in TR$. Therefore, it follows from the definition of the probability density function that the probability of success is

$$Y_{RT} = \Pr\left(\overline{\delta\mathbf{q}} \in TR\right) = \int_{TR} p_q\left(\overline{\delta\mathbf{q}}\right) d\overline{\delta\mathbf{q}} \tag{5.2.6}$$

where $TR = \left\{\overline{\delta\mathbf{q}} \,\middle|\, \left\|\overline{\delta\mathbf{q}}\right\|_\infty \le 1\right\}$. The advantage of this formulation is that the limits of integration are simple; Equation 5.2.6 can be expressed as

$$Y_{RT} = \int_{-1}^{1}\int_{-1}^{1}\cdots\int_{-1}^{1} p_q\left(\overline{\delta\mathbf{q}}\right) d\overline{\delta\mathbf{q}}_1 d\overline{\delta\mathbf{q}}_2 \cdots d\overline{\delta\mathbf{q}}_m \tag{5.2.7}$$

However, Equation 5.2.7 is not generally amenable to symbolic solution since the joint density function $p_q\left(\overline{\delta\mathbf{q}}\right)$ is not generally separable.

The solution to Equations 5.2.5 and 5.2.7 above will be the topic of the remainder of this chapter. I will develop analytical solutions and bounds for the special cases of greatest interest. I will also define algorithms that can be used to estimate rolled throughput yield in the most general case.

5.3 Single Quality Characteristic

Consider a process with a single quality characteristic. Naturally, there will be no effect of correlation on yield, so the scalar capability index C_p and bias factor k are appropriate measures of system performance. It is therefore valuable to relate the capability matrix $\mathbf{C}$ and bias vector $\mathbf{k}$ to these traditional capability metrics.

Theorem 5.3.1 If a product has exactly one quality characteristic (m=1), then the system has a bias factor and capability index of

$$k = \mathbf{k} \tag{5.3.1}$$

$$C_p = \frac{1}{\|\mathbf{C}\|_2} \tag{5.3.2}$$

Proof The capability matrix is defined as

$$\mathbf{C}_{ij} \equiv \frac{3\sigma_j \cdot \left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{ij}}{(U_i - L_i)/2} \tag{5.3.3}$$

If there is only one quality characteristic, this reduces to

$$\mathbf{C}_i = \frac{3\sigma_j \cdot \left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_i}{(U - L)/2} \tag{5.3.4}$$

Therefore, the normalized quality vector is

$$\overline{\delta \mathbf{q}} = \frac{3\sigma_j \cdot \left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_i}{(U - L)/2} \overline{\delta \mathbf{n}} + \mathbf{k} \tag{5.3.5}$$

The expected value of the quality characteristic is

$$E\left(\overline{\delta \mathbf{q}}\right) = E\left(\mathbf{C} \cdot \overline{\delta \mathbf{n}}\right) + \mathbf{k} \tag{5.3.6}$$

The expected value of a sum of random variables is the sum of the expected values. Also, the expected value of each element of the normalized noise vector $\overline{\delta \mathbf{n}}$ is zero, therefore

$$E\left(\overline{\delta \mathbf{q}}\right) = \mathbf{k} \tag{5.3.7}$$

By the definition of the normalized quality vector

$$E\left(\overline{\delta \mathbf{q}}\right) = \frac{E(\mathbf{q}) - \frac{U + L}{2}}{(U - L)/2} = k \tag{5.3.8}$$

which proves 5.3.1.

The standard deviation of the quality characteristic is

$$\sigma\left(\overline{\delta \mathbf{q}}\right) = \sigma\left(\mathbf{C} \cdot \overline{\delta \mathbf{n}} + \mathbf{k}\right) = \sigma\left(\mathbf{C} \cdot \overline{\delta \mathbf{n}}\right) \tag{5.3.9}$$

The standard deviation of a sum of random variables is the square root of the sum of the squared variances. The variance of each element of the normalized noise vector $\overline{\delta \mathbf{n}}$ is one third, therefore

$$\sigma\left(\overline{\delta\mathbf{q}}\right) = \sqrt{\sum_{j=1}^{n} \mathbf{C}_j \cdot \left(\frac{1}{3}\right)^2} = \frac{1}{3} \cdot \|\mathbf{C}\|_2 \tag{5.3.10}$$

From the definition of the normalized quality vector

$$\sigma\left(\overline{\delta\mathbf{q}}\right) = \frac{\sigma(\mathbf{q})}{(U-L)/2} \tag{5.3.11}$$

Combining the definition of the process capability index with the two expressions above

$$C_p \equiv \frac{(U-L)/2}{3 \cdot \sigma(\mathbf{q})} = \frac{1}{\|\mathbf{C}\|_2} \tag{5.3.12}$$

which proves 5.3.2.

In the special case of a single quality characteristic, the process capability index and bias vector completely define the first two moments of the distribution $p_q\left(\overline{\delta\mathbf{q}}\right)$. Therefore, it is possible to estimate the yield based on C_p and k. Since the normalized quality vector in this case degenerates to a scalar, the integration over the density function is one dimensional. The general expression of 5.2.7 becomes

$$Y_{RT} = \int_{-1}^{1} p_q\left(\overline{\delta\mathbf{q}}\right) d\overline{\delta\mathbf{q}} \tag{5.3.13}$$

There is no general solution to the above expression independent of the distributions of the noise vector elements. However, the quality characteristic can often be approximated as normally distributed even when the noise factors aren't normally distributed due to the Central Limit Theorem. This makes the following theorem of considerable practical value.

Theorem 5.3.2 If there is only one quality characteristic and it is normally distributed, then

$$Y_{RT} = \frac{1}{2}\left[\operatorname{erf}\left(\frac{3\sqrt{2}}{2} C_p (1-k)\right) + \operatorname{erf}\left(\frac{3\sqrt{2}}{2} C_p (1+k)\right)\right] \tag{5.3.14}$$

or, equivalently

$$Y_{RT} = \frac{1}{2}\left[\operatorname{erf}\left(\frac{3\sqrt{2}}{2} \frac{(1-\mathbf{k})}{\|\mathbf{C}\|_2}\right) + \operatorname{erf}\left(\frac{3\sqrt{2}}{2} \frac{(1+\mathbf{k})}{\|\mathbf{C}\|_2}\right)\right]. \tag{5.3.15}$$

Proof Given that the expected value of each element of the normalized noise vector $\overline{\delta \mathbf{n}}$ is zero, the expected value of the quality characteristic $E\left(\overline{\delta \mathbf{q}}\right) = k$. From the proof of Theorem 5.3.1,

$$\sigma\left(\overline{\delta \mathbf{q}}\right) = \frac{1}{3 \cdot C_p} \tag{5.3.16}$$

Substituting the form of the normal distribution into Equation 5.3.3 gives

$$\begin{aligned} Y &= \int_{-1}^{1} \frac{1}{\left(\frac{1}{3 \cdot C_p}\right)\sqrt{2\pi}} \exp\left[-\frac{(x-k)^2}{2 \cdot \left(\frac{1}{3 \cdot C_p}\right)^2}\right] dx \\ &= \frac{1}{2}\left[\operatorname{erf}\left(\frac{3\sqrt{2}}{2} C_p \cdot (1-k)\right) + \operatorname{erf}\left(\frac{3\sqrt{2}}{2} C_p \cdot (1+k)\right)\right] \end{aligned} \tag{5.3.17}$$

which proves the theorem.

It is interesting to note that the first term in Equation 5.3.14 contains the performance index C_{pk} ($C_{pk} = C_p(1-k)$). When this term dominates, yield is roughly a function of C_{pk}. Thus, Equation 5.3.14 reveals the theoretical basis for the usefulness of C_{pk} as an indicator of system performance.

The performance index C_{pk} accounts for both variation and bias. A low value of the metric may be caused by either poor centering of a distribution or by high spread. The advantage of the performance metric is that it provides a reasonable estimate of yield whether the measure is low due to variance or bias. This is revealed in Figures 5.3.1 and 5.3.2 which show that the defect rate ($1-Y_{RT}$) is strongly determined by the performance index regardless of the value of bias. At very low system performance levels ($C_{pk} < 1$) the performance index is not as predictive of defect rate. In these cases it is important to consider the separate effects of variance and bias. When the performance index is below one, it is important to discover the nature of the problem. It is important to know whether the cause is high variance, high bias, or both.

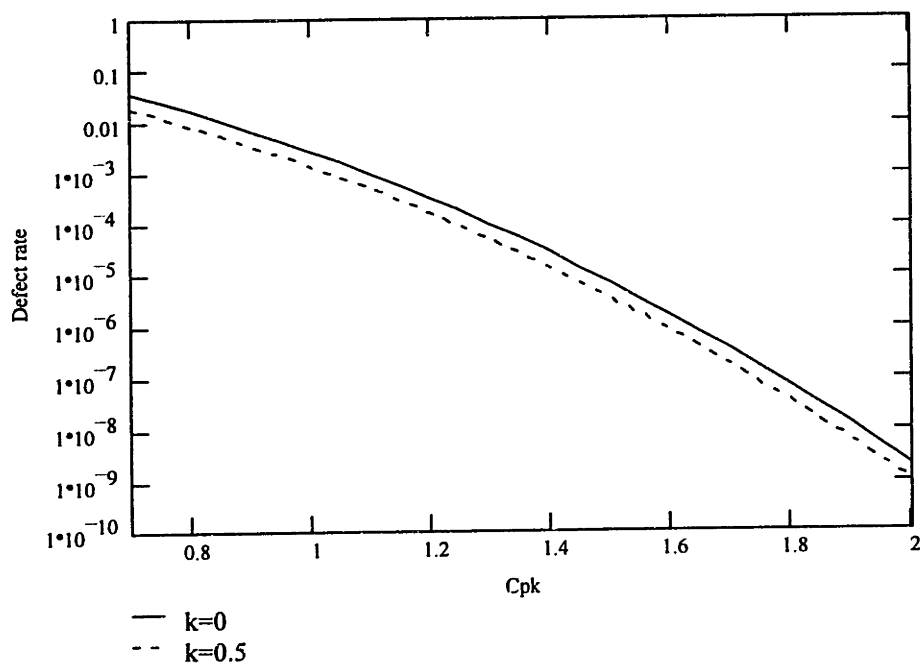


Figure 5.3.1 Log of the defect rate as a function of performance index and bias factor.

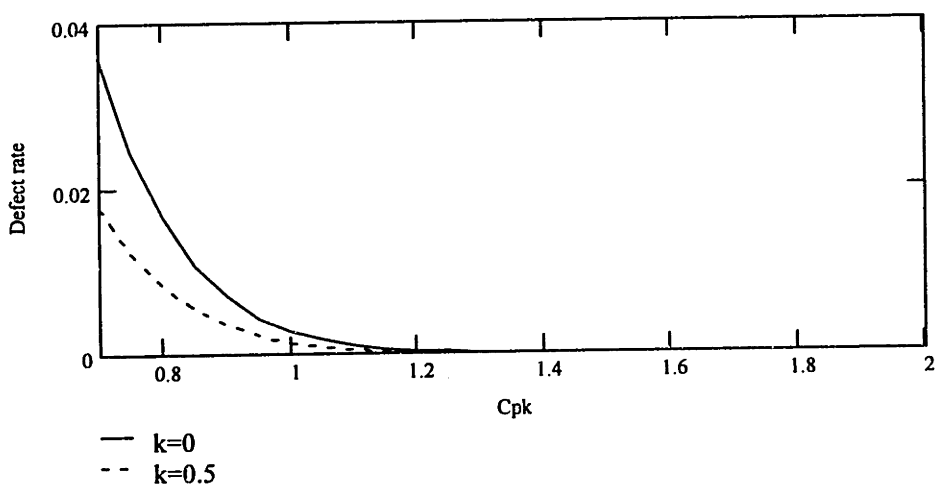


Figure 5.3.2 Defect rate as a function of performance index and bias factor.

As an example, consider the lead forming process. If we redefine the "product" to be single leads, then the system has only one quality characteristic as output – the side to side error in lead position. The data from an experimental run of nine 368 leaded MCMs is displayed in Figure 5.3.3 (The same data was displayed in Figure 4.5.2).

Based on this data, the performance index is quite low (C_{pk} =0.75). The poor performance is mainly due to high variance as evidenced by the low process capability (C_p =0.82). By comparison, the bias is not much of a contributor (k=0.08). This is consistent with the form of the histogram in Figure 5.3.3. This underscores the importance of reporting both C_p and k. If one were to report only the low C_{pk} value, one might consider actions to center the distribution, such as calibration of the assembly robot. However, it is clear that the performance index cannot be raised above 0.82 by calibration alone. As we shall see, the key to raising the performance of this system lies in removing the key sources of variance.

Given the capability index and bias factor for surface mount of MCMs, Equation 5.3.3 provides an estimate of the yield as 98.4%, which is rather poor. The physical interpretation is that, if an MCM is selected at random and a lead is selected at random from that MCM, the probability that it is within its tolerance range is 98.4%. This a consistent with the data set. Of the 3,312 leads measures, 98.3% were within their tolerance range. A binomial test at α=0.01 indicates that one cannot reject the hypothesis that these two probabilities are measures of the same population.

The predictive success of Equation 5.3.15 is due to the fact that the lead positions are approximately normally distributed as suggested by Figure 5.3.3. The normality of the distribution may be due to the fact that many different noise factors cause the observed variation. To the extent that no single noise factor dominates, one tends to observe a normal distribution in quality characteristics.

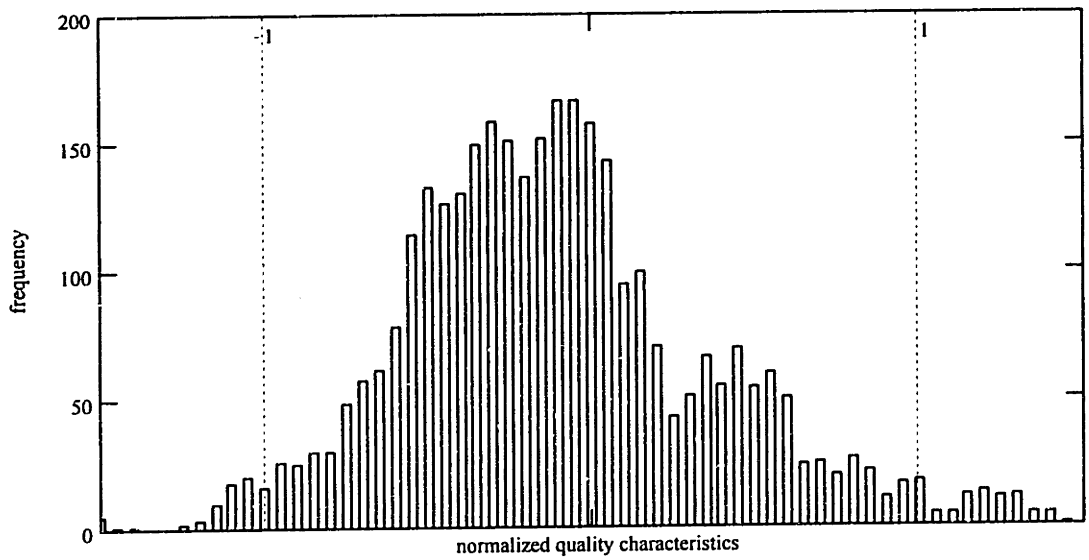


Figure 5.3.3 Data on lead position error from placement of nine 368 leaded MCMs – no adjustments applied.

One must use caution in applying Equation 5.3.14. A "vital few" noise factors often account for most of the variation. If a single noise factor dominates and that noise factor is not nearly normally distributed, then Equation 5.3.14 may be inaccurate.

This section has explored the special case that there is *only one* acceptance criterion. The following section introduces methods for products with multiple acceptance criteria.

5.4 The Monte Carlo Method

One of the greatest difficulties in computing an estimate of Y_{RT} is that manufacturing systems tend to be multi-input multi-output systems. If there are a large number of noise factors (n is large), then the strategy of integrating in $\overline{\delta \mathbf{n}}$ space demands a high dimensional integration. If there are a large number of quality characteristics (m is large), the strategy of integrating in $\overline{\delta \mathbf{q}}$ space demands a high dimensional integration. If both n and m are large, as in the MCM case study, then one needs an integration strategy well suited to high dimensional spaces. The Monte Carlo method is therefore an appropriate technique for analysis of manufacturing systems with many noise factors and quality characteristics.

The Monte Carlo method in its simplest form is defined by

$$P \cong \frac{1}{N} \sum_{trial=1}^{N} \xi_{trail} \tag{5.4.1}$$

where N is the number of trials performed and ξ is some event whose probability of occurrence, P, is to be estimated. This method is easily adapted to estimating rolled throughput yield of a manufacturing process. The event to be estimated is clearly defined in Equation 5.2.1. The algorithm is therefore

$$Y_{RT} \cong \frac{1}{\text{trials}} \sum_{k=1}^{\text{trials}} \left[\left\| \mathbf{C} \cdot \text{rand}(n,\ 1/3) + \mathbf{k} \right\|_{\infty} \leq 1 \right] \tag{5.4.2}$$

where rand(n, σ) is a function that returns a vector if n independent random variables selected from a population with a mean of zero and a standard deviation of σ.

The most significant feature of the Monte Carlo method is that its error δ in all cases is of the order

$$\delta = O\left(\frac{1}{\sqrt{N}}\right). \tag{5.4.3}$$

where N is the number of statistical trials [Buslenko, et. al., 1966]. This expression is more remarkable for what it omits than for what it includes. Specifically, it omits the dimensionality of the integral that is being evaluated. By contrast, the computational complexity of conventional quadrature formulae grow geometrically with the dimension of the problem.

The major difficulty with the Monte Carlo method is that the error bound lowers very slowly with increasing number of trials, N. To improve the accuracy by an additional decimal place, one must perform 100 times more statistical trials. Therefore, the Monte Carlo method is best suited to estimating the first two or three decimal places of a high dimensional problem. Methods to improve the accuracy of Monte Carlo integration, such as weighted sampling of the tails, will not be discussed in detail here.

The Monte Carlo method was applied to the case study on surface mount of MCMs. The process capability matrix and bias vector of Equations 4.4.1 and 4.4.2 were employed in the Monte Carlo algorithm of Equation 5.4.2. One hundred sets of subgroups of nine trials were performed. The confidence interval on rolled throughput yield was 64.4%-70.0% at α=0.05. This compares very favorably with the experimental data set. Of the nine MCMs formed and

measured, six would meet all tolerance specifications; The rolled throughput yield of the sample was 66.7%.

The close match between the performance of the data and the model suggests that the capability matrix approach to system design is not very sensitive to errors in modeling. Section 4.5 revealed that, although the mean and variance of the model and the data are closely matched, many of the correlation coefficients are not. The model only captured the a few major trends in the data (such as the linear trend in Figure 4.5.2). Section 8.5 further discusses the sensitivity of the results to modeling error.

The good performance of this capability matrix model is in stark contrast to the results of six sigma producibility analysis. As discussed in Section 5.3, the first time yield for this process is 98.4%. The six sigma approach is to multiply the first time yields to get an estimate of rolled throughput yield. Given a 368 leaded component, the estimate is therefore $0.984^{368} = 0.2\%$. Table 5.4.1 juxtaposes this result with that of the capability matrix approach and the results from the Hughes data.

Table 5.4.1 Comparison of rolled throughput yields – Hughes data, capability matrix model, and six sigma approach.

Hughes Data	Capability Matrix Model	Six Sigma Approach
66.7%	64.4%-70% @ α=0.05	0.2%

This section has developed a very general method that is well suited to finding rough (about 1 or 2 significant figures) estimates of rolled throughput yield based on the capability matrix and bias vector. The algorithm is appropriate for high dimensional problems such as the MCM surface mount case study. The next section develops numerical techniques appropriate for more accurate solutions to lower dimensional problems. The remaining sections will explore analytical solutions that apply under more restrictive assumptions.

5.5 Numerical Integration

The purpose of this section is to present a general expression for rolled throughput yield suitable for numerical computation by an appropriate deterministic numerical integration algorithm (e.g.

Simpson's rule). There shall be no restrictions on the form of the joint probability density function $p_n(\overline{\delta\mathbf{n}})$ or on the structure of the capability matrix **C** or bias vector **k**.

The event that all of the product specifications are simultaneously satisfied is the event that the normalized noise vector lies in the tolerance domain($\overline{\delta\mathbf{n}} \in TD$). As discussed in section 5.2, it follows that we may compute rolled throughput yield by integrating $p_n(\overline{\delta\mathbf{n}})$ over the TD

$$Y_{RT} = \int_{TD} p_n\left(\overline{\delta\mathbf{n}}\right) d\overline{\delta\mathbf{n}} \tag{5.5.1}$$

The tolerance domain is defined as $TD = \left\{ \overline{\delta\mathbf{n}} \,\middle|\, \left\| \mathbf{C} \cdot \overline{\delta\mathbf{n}} + \mathbf{k} \right\|_\infty \le 1 \right\}$. However, we require a means of expressing the integral over TD in a form suitable for numerical computations. The tolerance domain can be expressed as a system of linear inequalities.

$$TD = \left\{ \overline{\delta\mathbf{n}} \,\middle|\, \begin{bmatrix} \mathbf{C} \\ -\mathbf{C} \end{bmatrix} \cdot \overline{\delta\mathbf{n}} + \begin{Bmatrix} \mathbf{k} \\ -\mathbf{k} \end{Bmatrix} \le \begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix} \right\} \tag{5.5.2}$$

Each of the linear inequalities in Equation 5.5.2 above defines a half-space. The set of linear inequalities defines an n dimensional convex polyhedron in $\overline{\delta\mathbf{n}}$ space. To compute Y_{RT}, we need to integrate the joint density function of $\overline{\delta\mathbf{n}}$ over that convex polyhedron.

This raises a problem since the polyhedron is not necessarily bounded. Most deterministic methods of numerical integration require limits of integration, so some additional assumptions will be required. In practice, it is only necessary to integrate over the region $\left\{ \overline{\delta\mathbf{n}} \,\middle|\, \left\| \overline{\delta\mathbf{n}} \right\|_\infty \le 2 \right\}$. The restriction to a smaller region always causes the rolled throughput yield estimates to be slightly lower than the true Y_{RT} since probability density is a strictly positive quantity. If the elements of the noise vector are normally distributed, then the restriction to this smaller space will cause an error of no more than $2n$ ppb.

The rolled throughput yield can be computed by integrating over the polytope formed by the intersection of $\left\{ \overline{\delta\mathbf{n}} \,\middle|\, \left\| \overline{\delta\mathbf{n}} \right\|_\infty \le 2 \right\}$ and TD as expressed in Equation 5.5.2. The expression to be evaluated is

$$Y_{RT} = \int_{-2}^{2}\int_{-2}^{2}\cdots\int_{lb}^{ub} p\left(\overline{\delta\mathbf{n}}\right) d\overline{\delta\mathbf{n}}_1 d\overline{\delta\mathbf{n}}_2 \cdots d\overline{\delta\mathbf{n}}_n \tag{5.5.3}$$

where

$$ub = \min\left[\frac{1}{\mathbf{A}_{i,1}}\left(\mathbf{b}_i - \sum_{j=2}^{n}\mathbf{A}_{i,j}\overline{\delta\mathbf{n}}_{j-1}\right), \mathbf{A}_{i,1} > 0, i \in 1\ldots 2m\right] \tag{5.5.4}$$

and

$$lb = \min\left[\max\left[\frac{1}{\mathbf{A}_{i,1}}\left(\mathbf{b}_i - \sum_{j=2}^{n}\mathbf{A}_{i,1}\overline{\delta\mathbf{n}}_{j-1}\right), \mathbf{A}_{i,1} < 0, i \in 1\ldots 2m\right]\right] \tag{5.5.5}$$

$$ub$$

and

$$\mathbf{A} = \left[\frac{\mathbf{C}}{-\mathbf{C}}\right] \tag{5.5.6}$$

and

$$\mathbf{b} = \left\{\frac{\mathbf{1}_n}{\mathbf{1}_n}\right\} + \left\{\frac{-\mathbf{k}}{\mathbf{k}}\right\} \tag{5.5.7}$$

The upper and lower bounds of integration (*ub* and *lb*) as given by Equations 5.5.4 and 5.5.5 above ensure that the integration will not extend beyond *TD* as expressed in Equation 5.5.2. The upper and lower bounds define the points at which a line parallel to the axis $\overline{\delta\mathbf{n}}_1$ will intersect the convex polyhedron *TD*. This is depicted graphically for a simple 2D case in Figure 5.5.1.

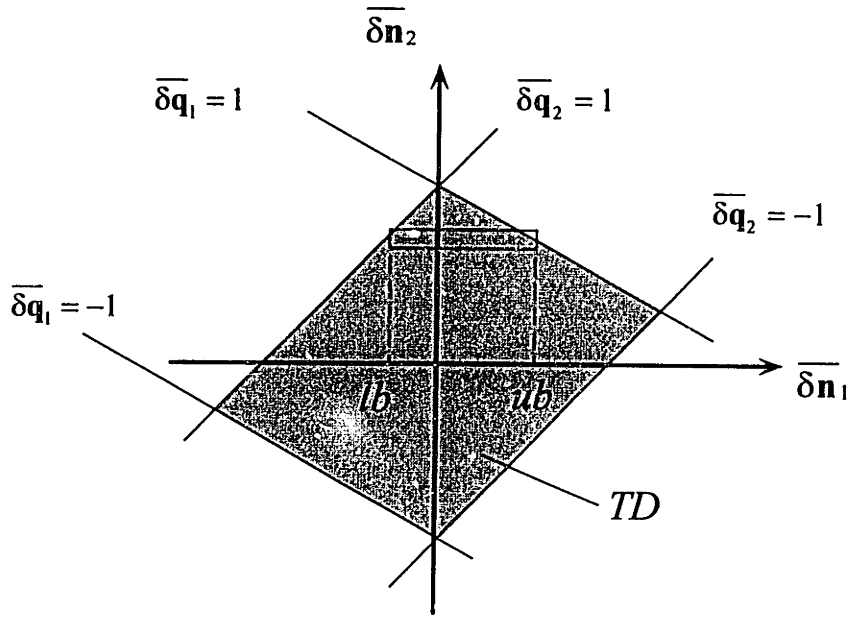


Figure 5.5.1 The limits of integration for numerical evaluation of rolled throughput yield.

Equations 5.4.3 through 5.4.7 represent an algorithm that can be incorporated into any suitable numerical integration scheme. However, the computational complexity tends to increase geometrically with the number of noise factors (computation time is $O(x^n)$).

As an example of the use of this algorithm, I will apply it to the surface mount of MCMs. The number of noise factors is far too great, so I will simplify the problem by considering only three noise factors. Assume that the lead frame is manufactured perfectly and that the placement robot performs perfectly. This leaves three of the most significant noise factors – the ones related to lead forming and assembly (noise factors 370, 371, and 372). The capability matrix and bias vector for the simplified system are

$$\mathbf{C} = \left[\begin{array}{c|c|c} 0.838 \cdot \mathbf{1}_{150} & 0.20 \cdot \mathbf{1}_{150} & \mathbf{0}_{150} \\ \hline 1.529 \cdot \mathbf{1}_{71} & \mathbf{0}_{71} & 0.20 \cdot \mathbf{1}_{71} \\ \hline 0.838 \cdot \mathbf{1}_{150} & -0.20 \cdot \mathbf{1}_{150} & \mathbf{0}_{150} \\ \hline 1.529 \cdot \mathbf{1}_{71} & \mathbf{0}_{71} & -0.20 \cdot \mathbf{1}_{71} \end{array}\right] \tag{5.5.8}$$

$$\mathbf{k} = \left\{\begin{array}{c} 0.1 \cdot \mathbf{1}_{150} \\ \hline 0.1 \cdot \mathbf{1}_{71} \\ \hline -0.1 \cdot \mathbf{1}_{150} \\ \hline -0.1 \cdot \mathbf{1}_{71} \end{array}\right\} \tag{5.5.9}$$

Applying the numerical computation algorithm to this **C** and **k** pair, one can compute rolled throughput yield regardless of the distribution of the noise factors. If the noise factors are assumed to be normally distributed, the resulting rolled throughput yield is 89%. The elimination of many of the noise factors caused the yield to be elevated above the value of the complete system as estimated by the Monte Carlo method in Section 5.4.

To demonstrate the power of the approach to deal with mixed distributions, I altered the problem and repeated the analysis. If noise factor #370 is switched to a uniform distribution, the resulting rolled throughput yield goes up to 98%. This is due to the shorter tails on the uniform distribution.

This section has developed a very general method. It applies to any distribution functions over the noise factors and includes no restrictions on the structure of the capability matrix and bias vector. The method is well suited to finding precise solutions for rolled throughput yield based on the capability matrix and bias vector. The algorithm is computationally tractable only for problems with relatively few noise factors. The remaining sections will explore closed form solutions that apply under more restrictive assumptions.

5.6 Normal Distributions

Normal distributions, or nearly normal distributions are common in manufacturing variation. This is a consequence of the Central Limit Theorem which states that the sum of a large number of finite random variables is normally distributed regardless of the distributions of the underlying random variables. This section will explore the case that all the noise factors are independent and normally distributed.

Before any substantive discussion, it will be convenient to form a shorthand notation for the joint normal distribution. Let $N(\mu,\mathbf{K})$ signify the joint normal distribution with a covariance matrix **K** and vector of means μ which is

$$p(\mathbf{x}) = \frac{1}{\left(\sqrt{2\pi}\right)^m \sqrt{|\mathbf{K}|}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{K}^{-1}(\mathbf{x}-\mu)\right\} \tag{5.6.1}$$

The elements of the normalized noise vector $\overline{\delta\mathbf{n}}_j$, by definition, have a mean of zero and a standard deviation of 1/3. If the noise factors are independent and normally distributed, then

they have a joint normal distribution of $N\left(0,\frac{1}{9}\mathbf{I}_{nxn}\right)$. The surfaces of constant probability density in $\overline{\delta\mathbf{n}}$ space will be n dimensional hyperspheres as shown in Figure 5.6.1.

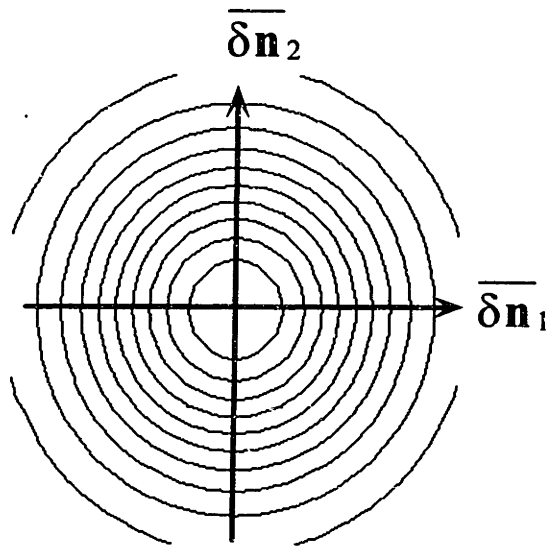


Figure 5.6.1 Lines of constant probability density if the noise factors are independent and normally distributed.

Theorem 5.6.1 If the normalized noise vector is composed of independent, normally distributed random variables $\overline{\delta\mathbf{n}}$ and the capability matrix has full row rank, then the joint distribution of the normalized quality vector $p\left(\overline{\delta\mathbf{q}}\right)$ is $N\left(\mathbf{k},\frac{1}{9}\mathbf{C}\cdot\mathbf{C}^T\right)$.

Proof A linear transformation by a matrix $\mathbf{C}$ of a joint normal random vector with covariance $\mathbf{K}$ is itself joint normal with covariance $\mathbf{CKC}^T$ [Clifford, 1973]. The normalized noise vector $\overline{\delta\mathbf{n}}$ has a joint normal distribution of $N\left(0,\frac{1}{9}\mathbf{I}_{nxn}\right)$. By definition, $\overline{\delta\mathbf{q}} = \mathbf{C}\cdot\overline{\delta\mathbf{n}} + \mathbf{k}$. Therefore, $\overline{\delta\mathbf{q}}$ is normally distributed with covariance matrix is $\mathbf{C}\cdot\frac{1}{9}\mathbf{I}\cdot\mathbf{C}^T = \frac{1}{9}\mathbf{C}\cdot\mathbf{C}^T$. The expected value of $\overline{\delta\mathbf{q}}$ is

$$E\left(\overline{\delta\mathbf{q}}\right) = E\left(\mathbf{C}\cdot\overline{\delta\mathbf{n}}\right) + \mathbf{k} \tag{5.6.2}$$

Since the expected values of the normalized noise vector $\overline{\delta \mathbf{n}}_j$ are zero, the expected value of $\overline{\delta \mathbf{q}}$ is **k**.

When the vector is jointly normally distributed, the lines of constant probability density are m dimensional ellpsoids as shown in Figure 5.6.2. In general, the elements of $\overline{\delta \mathbf{q}}$ are not probabilistically independent in which case the major axes of the ellipsoids of constant probability density will not be aligned with the axes of $\overline{\delta \mathbf{q}}$ space as shown in Figure 5.6.2.

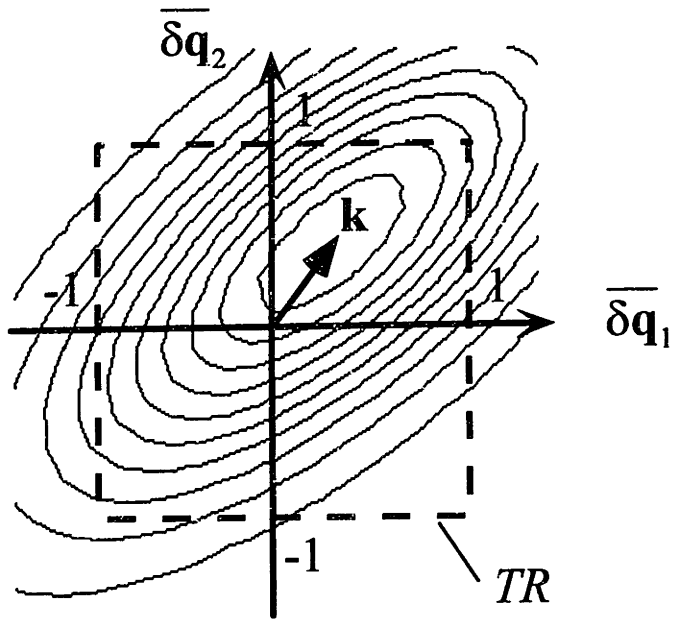


Figure 5.6.2 The space of normalized quality characteristics with a joint normal distribution.

Theorem 5.6.1 leads to a reduction rule for process capability matrices as expressed in the following Lemma.

Lemma 5.6.1 For normally distributed noise factors, if the pth and q^{th} column vectors of a capability matrix are parallel, then the n by m capability matrix **C** may be reduced to an n-1 by m matrix **C'** without affecting the joint density function over the quality characteristics where

$$
\begin{aligned}
&\mathbf{C}'_{ij} = \mathbf{C}_{ij} && \text{if } j < p \text{ or } p < j < q \\
&\mathbf{C}'_{ij} = \sqrt{\left(\mathbf{C}_{ip}\right)^2 + \left(\mathbf{C}_{iq}\right)^2} && \text{if } j = p \\
&\mathbf{C}'_{i(j-1)} = \mathbf{C}_{ij} && \text{if } j > q \\
&\text{for all } i \in 1 \ldots m,\ j \in 1 \ldots n
\end{aligned}
\tag{5.6.3}
$$

and the bias vector **k** remains unchanged.

Proof By the definition of matrix multiplication and transposition

$$\left(\mathbf{C}\mathbf{C}^T\right)_{ij} = \sum_{k=1}^{n} \mathbf{C}_{ik}\mathbf{C}_{jk} \tag{5.6.4}$$

Pulling the p and q terms from the summation

$$\left(\mathbf{C}\mathbf{C}^T\right)_{ij} = \left[\sum_{\substack{k=1 \\ k \neq p,q}}^{n} \mathbf{C}_{ik}\mathbf{C}_{jk}\right] + \mathbf{C}_{ip}\mathbf{C}_{jp} + \mathbf{C}_{iq}\mathbf{C}_{jq} \tag{5.6.5}$$

Expanding the last term

$$\left(\mathbf{C}\mathbf{C}^T\right)_{ij} = \left[\sum_{\substack{k=1 \\ k \neq p,q}}^{n} \mathbf{C}_{ik}\mathbf{C}_{jk}\right] + \sqrt{\mathbf{C}_{ip}{}^2\mathbf{C}_{jp}{}^2 + 2\mathbf{C}_{ip}\mathbf{C}_{jp}\mathbf{C}_{iq}\mathbf{C}_{jq} + \mathbf{C}_{iq}{}^2\mathbf{C}_{jq}{}^2} \tag{5.6.6}$$

If the pth and q^{th} column vectors of a capability matrix are parallel then

$$2\mathbf{C}_{ip}\mathbf{C}_{jp}\mathbf{C}_{iq}\mathbf{C}_{jq} = \mathbf{C}_{iq}{}^2\mathbf{C}_{jp}{}^2 + \mathbf{C}_{ip}{}^2\mathbf{C}_{jq}{}^2 \tag{5.6.7}$$

Substituting 5.6.7 into 5.6.6 and simplifying

$$\left(\mathbf{C}\mathbf{C}^T\right)_{ij} = \left[\sum_{\substack{k=1 \\ k \neq p,q}}^{n} \mathbf{C}_{ik}\mathbf{C}_{jk}\right] + \sqrt{\mathbf{C}_{ip}{}^2 + \mathbf{C}_{iq}{}^2}\sqrt{\mathbf{C}_{jp}{}^2 + \mathbf{C}_{jq}{}^2} \tag{5.6.8}$$

Substituting Equation 5.6.3

$$\left(\mathbf{C}\mathbf{C}^T\right)_{ij} = \left[\sum_{\substack{k=1 \\ k \neq p}}^{n-1} \mathbf{C}'_{ik}\,\mathbf{C}'_{jk}\right] + \mathbf{C}'_{ip}\,\mathbf{C}'_{jp} = \left(\mathbf{C}'\mathbf{C}'^T\right)_{ij} \tag{5.6.9}$$

According to theorem 5.6.1, the probability density function over the normalized quality characteristics is $N\left(\mathbf{k},\frac{1}{9}\mathbf{C}\cdot\mathbf{C}^T\right)$. Since neither of the arguments of N change upon the reduction of $\mathbf{C}$ to $\mathbf{C}'$, the joint density function is unchanged. ▒

Theorem 5.6.1, shows that if $\overline{\delta\mathbf{n}}$ is independent and normally distributed, then the density function over $\overline{\delta\mathbf{q}}$ is known explicitly as a function of $\mathbf{C}$ and $\mathbf{k}$. Therefore, rolled throughput yield can be computed by integrating over the tolerance range. Combining Equation 5.2.7 and 5.6.1 gives

$$Y_{RT}=\int_{-1}^{1}\int_{-1}^{1}\cdots\int_{-1}^{1}\frac{9}{\left(\sqrt{2\pi}\right)^m\sqrt{\left|\mathbf{C}\cdot\mathbf{C}^T\right|}}\exp\left\{-\frac{9}{2}(\overline{\delta\mathbf{q}}-\mathbf{k})^T\left[\mathbf{C}\cdot\mathbf{C}^T\right]^{-1}(\overline{\delta\mathbf{q}}-\mathbf{k})\right\}d\overline{\delta\mathbf{q}}_1 d\overline{\delta\mathbf{q}}_2\cdots d\overline{\delta\mathbf{q}}_m \quad (5.6.10)$$

Equation 5.6.10 does not generally admit a closed form solution. However, if there is no bias, then there exist rigorous upper and lower limits on the rolled throughput yield in the limit of a large numbers of trials

$$\prod_{i=1}^{m}\mathrm{erf}\left(\frac{3\sqrt{2}}{2\sqrt{\sum_{j=1}^{n}\mathbf{C}_{i,j}^{\ 2}}}\right)\le Y_{RT}\le\prod_{i=1}^{m}\mathrm{erf}\left(\frac{3\sqrt{2}}{2\cdot\mathbf{D}_{ii}\cdot\max(\mathbf{U}_i)}\right)\ \text{if}\ \mathbf{k}=0 \quad (5.6.11)$$

where the singular value decomposition of $\mathbf{C}$ is $\mathbf{UDV}^T$ and $\mathbf{U}_i$ is the ith row vector of the orthogonal matrix $\mathbf{U}$.

If there are relatively few quality characteristics, then Equation 5.6.10 is useful for numerical computations. For example, if we wish to consider the surface mount of MCMs, we might reduce the computational complexity of the problem by considering only three of the leads, say #1, #125, and #184 (for numbering scheme, see Figure 5.6.3). We may compute the probability that these three leads meet their tolerances by using the reduced capability matrix and bias vector

$$\mathbf{C}=\begin{bmatrix}0.35 & 0 & 0 & -0.02 & 0.838 & 0.231 & 0.22 & 0\\ 0 & 0.35 & 0 & 0.02 & 0.838 & 0.231 & 0.22 & 0\\ 0 & 0 & 0.35 & 0.01 & 1.529 & 0.462 & 0 & 0.22\end{bmatrix} \quad (5.6.12)$$

$$\mathbf{k} = \begin{Bmatrix} -0.1 \\ 0.3 \\ 0.2 \end{Bmatrix} \tag{5.6.13}$$

Using Equation 5.6.2 to compute the rolled throughput yield for the system described by the C and k in 5.6.5 and 5.6.6 gives $Y_{RT} = 90\%$. This is somewhat higher that that for the full system which is to be expected since there are fewer opportunities for non-conformance in this reduced system.

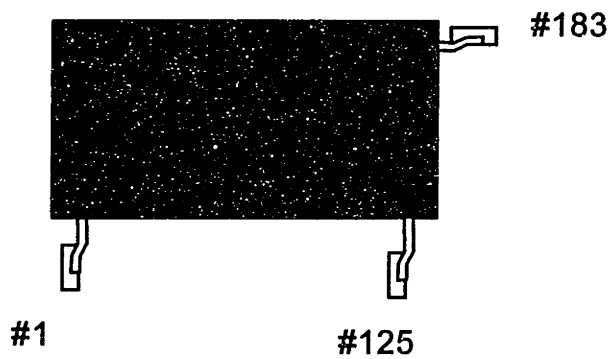


Figure 5.6.3 An MCM with only three leads used in a simplified example problem.

This section has presented equations that may be used for numerical integration of rolled throughput yield when the noise factors are independent and normally distributed. This approach works only when there are relatively few quality characteristics. When there are a large number of quality characteristics, but the noise factors are normally distributed, some other approaches should be employed. The next section considers a special case in which the quality characteristics are independent.

5.7 Independent, Normal Quality Characteristics

An important consequence of Theorem 5.6.1 is that even though the $\overline{\delta \mathbf{n}}_j$ were assumed to be probabilistically independent, the normalized quality characteristics $\overline{\delta \mathbf{q}}_i$ are often not probabilistically independent. This point is clarified in the following Lemma of Theorem 5.6.1.

Lemma 5.7.1 If the normalized noise vector is composed of independent, normally distributed random variables $\overline{\delta\mathbf{n}}$, then the normalized quality characteristics $\overline{\delta\mathbf{q}_i}$ are probabilistically independent iff the row vectors of **C** are mutually orthogonal.

Proof If a vector $\overline{\delta\mathbf{q}}$ has a joint normal distribution $N(\mu,\mathbf{K})$, then its ith and j^{th} elements are probabilistically independent iff $\mathbf{K}_{ij} = 0$ for all $i \neq j$ [Cover and Thomas, 1991]. From Theorem 5.6.1, $p\left(\overline{\delta\mathbf{q}}\right) = N\left(\mathbf{k}, \frac{1}{9}\mathbf{C}\cdot\mathbf{C}^T\right)$. The off diagonal elements of the covariance matrix will be zero iff $\mathbf{C}\cdot\mathbf{C}^T = 0$ for all $i \neq j$, that is, iff the row vectors of **C** are mutually orthogonal.

This lemma leads to an exact, closed form expression for rolled throughput yield in the special case that the row vectors of **C** are mutually orthogonal.

Theorem 5.7.2 If the normalized noise vector is composed of independent, normally distributed random variables $\overline{\delta\mathbf{n}}$ and the row vectors of **C** are mutually orthogonal then

$$Y_{RT} = \prod_{i=1}^{m} \frac{1}{2}\left[\operatorname{erf}\left(\frac{3\sqrt{2}}{2}\frac{(1-\mathbf{k}_i)}{\|\mathbf{C}_i\|_2}\right) + \operatorname{erf}\left(\frac{3\sqrt{2}}{2}\frac{(1+\mathbf{k}_i)}{\|\mathbf{C}_i\|_2}\right)\right] \tag{5.7.1}$$

where $\|\mathbf{C}_i\|_2$ denotes the Euclidean norm of the ith row vector of the capability matrix.

Proof By Lemma 5.7.1, the elements of the quality characteristic vector $\overline{\delta\mathbf{q}}$ are probabilistically independent. The assumption of statistical independence allows one to compute Y_{RT} by multiplying the probabilities of satisfying each quality characteristic

$$Y_{RT} = \prod_{i=1}^{m}\left[\int_{-1}^{1} p\left(\overline{\delta\mathbf{q}_i}\right) d\overline{\delta\mathbf{q}_i}\right] \tag{5.7.2}$$

The expected value of a sum of random variables is the sum of the expected values. Also, the expected value of each element of the normalized noise vector $\overline{\delta\mathbf{n}}$ is zero, therefore

$$E\left(\overline{\delta \mathbf{q}_i}\right) = \mathbf{k}_i \tag{5.7.3}$$

The variance of each element of the normalized noise vector $\overline{\delta \mathbf{n}}$ is one third, therefore

$$\sigma\left(\overline{\delta \mathbf{q}_i}\right) = \sqrt{\sum_{j=1}^{n} \mathbf{C}_{ij} \cdot \left(\frac{1}{3}\right)^2} = \frac{1}{3} \cdot \|\mathbf{C}_i\|_2 \tag{5.7.4}$$

Substituting the form of the normal distribution into 5.7.4 gives

$$\begin{aligned} Y &= \prod_{i=1}^{m} \left[\int_{-1}^{1} \frac{1}{\left(\frac{1}{3}\|\mathbf{C}_i\|_2\right)\sqrt{2\pi}} \exp\left[-\frac{(\overline{\delta \mathbf{q}_i} - \mathbf{k}_i)^2}{2 \cdot \left(\frac{1}{3}\|\mathbf{C}_i\|_2\right)^2} \right] d\overline{\delta \mathbf{q}_i} \right] \\ &= \prod_{i=1}^{m} \frac{1}{2} \left[\operatorname{erf}\left(\frac{3\sqrt{2}}{2} \cdot \frac{(1-\mathbf{k}_i)}{\|\mathbf{C}_i\|_2} \right) + \operatorname{erf}\left(\frac{3\sqrt{2}}{2} \cdot \frac{(1+\mathbf{k}_i)}{\|\mathbf{C}_i\|_2} \right) \right] \end{aligned} \tag{5.7.5}$$

which proves the theorem.

The results of this theorem may be applied to the surface mount of MCMs. Let us take the system that is characterized by the capability matrix and bias vector of Equations 4.4.1 and 4.4.2. Let us further assume that the noise factors are independent and normally distributed. As evidenced by taking the dot products of any two row vectors in this matrix, there is correlation among the quality characteristics. If one neglects the correlation and employs Equation 5.7.1, the resulting estimate of rolled throughput yield is 0.2%. This is in strong contrast to the α=0.05 range of 64.4%-70% derived from Monte Carlo simulations.

This result of the computation above has considerable industrial significance. The case study on surface mount of MCMs is not atypical of problems that manufacturing engineers routinely face. The solution strategy offered by the most widely accepted producibilty analysis method, six sigma methodology, suggests neglecting correlation among the quality characteristics [Harry and Lawson, 1992]. The computation shown above indicates that this assumption leads to a gross error in computation (which may lead to poor decision making). In other words, there is a substantial weakness in the state of the art in producibility analysis which the algorithms and equations of this chapter address very effectively.

This section concludes the examination of normal distributions. Another important distribution function for noise is the uniform distribution. These distributions are examined in light of capability matrix models in the following section.

5.8 Uniform Distributions

This section explores cases in which all of the probability density functions of the process variables are uniformly distributed and there are no restrictions on the form of the capability matrix **C** and bias vector **k**. The section introduces a geometric representation of the problem and an algorithm for computing rolled throughput yield.

For uniformly distributed, statistically independent $\overline{\delta\mathbf{n}}$, the probability density is uniformly distributed within an n dimensional hyper-cube in $\overline{\delta\mathbf{n}}$ space. The hypercube in three dimensional $\overline{\delta\mathbf{n}}$ space (a plain old cube in this case) is depicted in Figure 5.8.1. Each side of the cube is $2/\sqrt{3}$ units long, the hypercube is centered at the origin, and every face of the hypercube is normal to an axis in $\overline{\delta\mathbf{n}}$ space. Because the probability density is zero everywhere outside this hypercube, the hypercube is the support set S of the random vector $\overline{\delta\mathbf{n}}$. The support set can be expressed as a system of linear inequalities

$$S = \left\{ \overline{\delta\mathbf{n}} \;\middle|\; \begin{bmatrix} \mathbf{I}/\sqrt{3} \\ -\mathbf{I}/\sqrt{3} \end{bmatrix} \cdot \overline{\delta\mathbf{n}} \le \begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix} \right\} \tag{5.8.1}$$

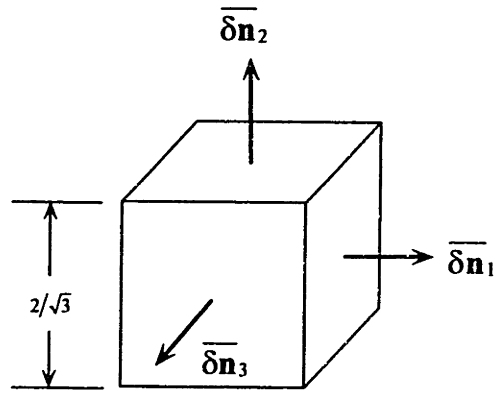


Figure 5.8.1 The support set S given uniformly distributed noise factors.

As discussed in section 5.2, the rolled throughput yield may be computed by integrating over the tolerance range TD using Equation 5.2.4. In this case, since a support set has been defined, it is only necessary to integrate over the intersection of TD and S

$$Y_{RT} = \int_{TD \cap S} p\left(\overline{\delta \mathbf{n}}\right) d\overline{\delta \mathbf{n}} \tag{5.8.2}$$

Taking the definition of the tolerance domain (Equation 5.5.2) and combining it with the definition of the support set we find

$$TD \cap S = \left\{ \overline{\delta \mathbf{n}} \;\middle|\; \begin{bmatrix} \mathbf{C} \\ -\mathbf{C} \\ \mathbf{I}/\sqrt{3} \\ -\mathbf{I}/\sqrt{3} \end{bmatrix} \cdot \overline{\delta \mathbf{n}} + \begin{Bmatrix} \mathbf{k} \\ -\mathbf{k} \\ 0 \\ 0 \end{Bmatrix} \leq \begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix} \right\} \tag{5.8.3}$$

The set defined in Equation 5.8.3 is the intersection of a hyper-cube and a convex polyhedron, therefore it is a polytope (a bounded polyhedron). Figure 5.8.2 provides a graphical depiction of the concept in two dimensions.

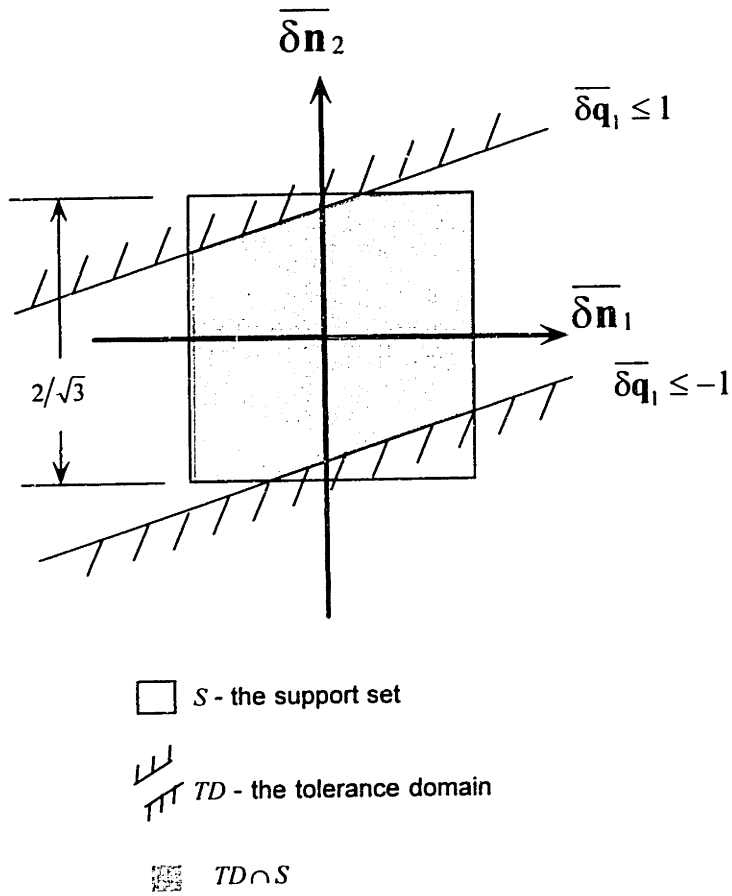


Figure 5.8.2 The intersection of the support set and tolerance domain is a polytope.

Because we are dealing with a uniform distribution, the density function $p_n\left(\overline{\delta \mathbf{n}}\right)$ is a constant. The integral over the entire support set of any probability density function must equal one. This fact, combined with the definition of the support set (Equation 5.10.1) implies that the density function must be

$$p_n\left(\overline{\delta \mathbf{n}}\right) = \begin{cases} \dfrac{1}{\left(2\sqrt{3}\right)^n} & \text{if } \left\|\overline{\delta \mathbf{n}}\right\|_\infty \le 1 \\ 0 & \text{otherwise} \end{cases} \tag{5.8.4}$$

The rolled throughput yield for the case of uniformly distributed $\overline{\delta \mathbf{n}}$ has a geometric interpretation. Y_{RT} is the product of the volume of the polytope $TD \cap S$ and the probability density within the support set S which is everywhere $1/\left(2\sqrt{3}\right)^n$.

$$Y_{RT} = \frac{V(TD \cap S)}{\left(2\sqrt{3}\right)^n} \tag{5.8.5}$$

Therefore, in order to compute the rolled throughput yield, we require an algorithm to compute the volumes in Equation 5.8.5 above. I developed such an algorithm based on a theorem discovered by Lasserre [1981]. The algorithm is documented in Appendix G. The computational complexity of the algorithm is strongly dependent on the dimension of the polytope n and on the number of active constraints [Lasserre, 1981].

The algorithm of this section can be applied to the surface mount problem. It is useful to reduce the dimension of the problem since the computational complexity of the algorithm is $O(mn)$. Following the procedure used in Section 5.6 we consider only three of the leads, (#1, #125, and #184). The reduced capability matrix and bias vector are

$$\mathbf{C} = \begin{bmatrix} 0.35 & 0 & 0 & -0.02 & 0.838 & 0.231 & 0.22 & 0 \\ 0 & 0.35 & 0 & 0.02 & 0.838 & 0.231 & 0.22 & 0 \\ 0 & 0 & 0.35 & 0.01 & 1.529 & 0.462 & 0 & 0.22 \end{bmatrix} \tag{5.8.6}$$

$$\mathbf{k} = \begin{Bmatrix} -0.1 \\ 0.3 \\ 0.2 \end{Bmatrix} \tag{5.8.7}$$

Using the algorithm in Appendix G, the rolled throughput yield for the system is $Y_{RT} =$ 92%. This is somewhat higher that that for the full system which is to be expected since there are fewer opportunities for non-conformance in this reduced system. It is comparable to that computed for the same system with normally distributed noise factors (see Section 5.6) though somewhat higher due to the lack of tails on the uniform distributions.

This section has presented equations that may be used for numerical integration of rolled throughput yield when the noise factors are independent and uniformly distributed. This approach is most appropriate in systems with relatively few quality characteristics and noise factors. The algorithms do, however, provide solutions whose accuracy is limited only by floating point error. The next section summarizes the results of this chapter.

5.9 Chapter Conclusions

This chapter has developed ways to compute rolled throughput yield based on the capability matrix and bias vector. The rolled throughput yield (Y_{RT}) is defined as the probability that a discrete unit of product conforms to *all* of its tolerances. It can also be defined in terms of the capability matrix and bias vector. This definition spawned the set of analytical tools for computing Y_{RT} listed in Table 5.9.1. The method most appropriate to a given problem is a function of several factors:

- The form of density function over the noise factors
- The structure of the capability matrix **C** and bias factor **k**
- The dimensionality of the problem as determined by the number of quality characteristics m and number of noise factors n
- The desired accuracy of the solution

If there is only one quality characteristic, then the traditional capability index and bias factor are sufficient to capture the system behavior. If there are many noise factors, then the quality characteristic will tend to be normally distributed. In this case, there is an exact closed form solution for Y_{RT}.

The Monte Carlo algorithm is very general in its application, the noise factors can have any distributions and there are no restrictions on **C** and **k**. Its accuracy grows very slowly with number of trials, but is not dependent on the dimensionality of the problem. Therefore, it is a good choice for rough solutions to high dimensional problems.

Numerical integration over the space of noise factors is a very general approach. It allows one to consider any type of distribution of the noise and it provides good accuracy in most cases. However, the computational complexity grows geometrically with number of noise factors (n).

If the probability density over the quality characteristics is jointly Gaussian (or nearly so) then the probability density function can be expressed as a function of **C** and **k**. There is generally not a closed form solution for this integral. However, if there are relatively few quality characteristics, the numerical integration over the quality characteristics is tractable.

If the quality characteristics are jointly Gaussian and there is no bias, then there exist closed form bounds on the solution. In some cases, the bounds will be tightly spaced so that the bounds alone provide sufficient accuracy. In other cases, they can be useful as a check on other techniques.

If the quality characteristics are jointly Gaussian, one can test for independence of the quality characteristics using the capability matrix. If the quality characteristics are independent, then a closed form solution exists for Y_{RT}.

If the quality characteristics are uniformly distributed, then there exists a geometric interpretation of rolled throughput yield. A recursive algorithm for computing the volume of a polytope will provide a value for Y_{RT}.

Among the most important conclusions to be derived from this chapter concern the performance of the above algorithms and equations on the surface mount of MCMs. If the capability matrix and bias vector are indeed a reasonable model of the manufacturing system, then the system and model should behave similarly in certain important respects. In this chapter I showed that the tools listed above provide excellent estimates of rolled throughput yield when applied properly. However, if correlation among the quality characteristics is ignored, the estimates of Y_{RT} can be in error by orders of magnitude. This reveals some of the limitations in current approaches to producibility analysis such as six sigma methodology.

The MCM surface mount process modeled in this chapter has a rolled throughput yield of only 2/3. To make the system economically viable, rolled throughput yield should be improved by some means. In the next chapter, I consider ways to improve yield by tightening product specifications and/or loosening tolerances. In Chapter 7, I consider improving yield using on-line adjustment procedures.

Table 5.9.1 Equations for Estimating Yield Based on Process Capability Indices

Definition	$\Pr(L_i \le \mathbf{q}_i \le U_i)$ for all $i \in 1, 2, \ldots m$
As a function of **C** and **k**	$Y_{RT} = \Pr\left(\left\|\mathbf{C}\cdot\overline{\delta\mathbf{n}} + \mathbf{k}\right\|_\infty \le 1\right)$ where $E\left(\overline{\delta\mathbf{n}}_j\right) = 0$ and $\sqrt{E\left(\left[\overline{\delta\mathbf{n}}_j - E\left(\overline{\delta\mathbf{n}}_j\right)\right]^2\right)} = \frac{1}{3}$ for all $j \in 1\ldots n$
Single quality characteristic	$\lim_{n\to\infty} Y = \frac{1}{2}\left[\operatorname{erf}\left(\frac{3\sqrt{2}}{2}C_p(1-k)\right) + \operatorname{erf}\left(\frac{3\sqrt{2}}{2}C_p(1+k)\right)\right]$
Monte Carlo algorithm	$Y_{RT} \cong \frac{1}{\text{trials}}\sum_{k=1}^{\text{trials}}\left[\left\|\mathbf{C}\cdot\operatorname{rand}(n,\ 1/3) + \mathbf{k}\right\|_\infty \le 1\right]$
Numerical integration	$Y_{RT} = \int_{TD} p\left(\overline{\delta\mathbf{n}}\right)d\overline{\delta\mathbf{n}}$ where $TD = \left\{\overline{\delta\mathbf{n}} \,\middle\|\, \begin{bmatrix}\mathbf{C}\\ -\mathbf{C}\end{bmatrix}\cdot\overline{\delta\mathbf{n}} + \begin{Bmatrix}\mathbf{k}\\ -\mathbf{k}\end{Bmatrix} \le \begin{Bmatrix}1\\1\\\vdots\\1\end{Bmatrix}\right\}$
Jointly Gaussian	$Y_{RT} = \int_{-1}^{1}\int_{-1}^{1}\cdots\int_{-1}^{1}\frac{3}{\left(\sqrt{2\pi}\right)^m\sqrt{\left\|\mathbf{C}\cdot\mathbf{C}^T\right\|}}\exp\left\{-\frac{9}{2}(\overline{\delta\mathbf{q}}-\mathbf{k})^T\left[\mathbf{C}\cdot\mathbf{C}^T\right]^{-1}(\overline{\delta\mathbf{q}}-\mathbf{k})\right\}d\overline{\delta\mathbf{q}}_1 d\overline{\delta\mathbf{q}}_2\cdots d\overline{\delta\mathbf{q}}_m$
Jointly Gaussian with no bias	$\prod_{i=1}^{m}\operatorname{erf}\left(\frac{3\sqrt{2}}{2\sqrt{\sum_{j=1}^{n}\mathbf{C}_{i,j}^{\ 2}}}\right) \le Y_{RT} \le \prod_{i=1}^{m}\operatorname{erf}\left(\frac{3\sqrt{2}}{2\cdot\mathbf{D}_{ii}\cdot\max(\mathbf{U}_i)}\right)$ if $\mathbf{k} = 0$
Jointly Gaussian and uncorrelated	$\prod_{i=1}^{m}\frac{1}{2}\left[\operatorname{erf}\left(\frac{3\sqrt{2}}{2}\frac{(1-\mathbf{k}_i)}{\sqrt{\sum_{j=1}^{n}\mathbf{C}_{ij}^{\ 2}}}\right) + \operatorname{erf}\left(\frac{3\sqrt{2}}{2}\frac{(1+\mathbf{k}_i)}{\sqrt{\sum_{j=1}^{n}\mathbf{C}_{ij}^{\ 2}}}\right)\right]$
Uniform	$Y_{RT} = \frac{V(TD\cap S)}{\left(2\sqrt{3}\right)^n}$ where $TD\cap S = \left\{\overline{\delta\mathbf{n}} \,\middle\|\, \begin{bmatrix}\mathbf{C}\\ -\mathbf{C}\\ \mathbf{I}/\sqrt{3}\\ -\mathbf{I}/\sqrt{3}\end{bmatrix}\cdot\overline{\delta\mathbf{n}} + \begin{Bmatrix}\mathbf{k}\\ -\mathbf{k}\\ 0\\ 0\end{Bmatrix} \le \begin{Bmatrix}1\\1\\\vdots\\1\end{Bmatrix}\right\}$

6. Setting Tolerances and Specifications

The goal of this thesis is to help engineers design manufacturing systems whose products conform to their dimensional tolerances. The last chapter presented many ways to evaluate the ability of a proposed process to perform its function as measured by rolled throughput yield. This chapter considers ways to improve yield by changing certain parameters of the system.

Given the definition of the elements of the process capability matrix

$$\mathbf{C}_{ij} \equiv \frac{3\sigma_j \cdot \left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{ij}}{(U_i - L_i)/2} \tag{6.1}$$

three strategies arise naturally:

- Widen the tolerance on some of the dimensions of the product – that is, increase some of the $(U_i - L_i)$.
- Change the Jacobian matrix $\left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{ij}$ of the manufacturing system (e.g. by changing the process set point, **t**).
- Reduce the variance of some of the noise factors – that is, reduce some of the σ_j

6.1 Widening Tolerances

In this section, we will explore the effects of widening the tolerances on rolled throughput yield. Presumably, widening tolerances can induce costs due to the lower average satisfaction of the customer. However, in some cases, this cost may be more than compensated by increased yield. To make a decision on cost optimal tolerance specifications, it is useful to have a means of computing the sensitivity of the rolled throughput yield (Y_{RT}) to each tolerance width.

To begin this section, it is essential to determine the effect of a change in tolerance width on the capability matrix and bias vector.

Theorem 6.1.1 If a system has a capability matrix **C** and bias vector **k** and the pth tolerance width is widened by a fraction ε, then the new capability matrix **C'** and bias vector **k'** will be

$$\mathbf{C}' = \mathbf{TC} \tag{6.1.1}$$

and

$$\mathbf{k}' = \mathbf{Tk} \tag{6.1.2}$$

where **T** is an m by m matrix such that

$$\mathbf{T}_{ij} = \begin{cases} 0 & \text{if} \quad i \neq j \\ \dfrac{1}{1+\varepsilon} & \text{if} \quad i = p = j \\ 1 & \text{otherwise} \end{cases} \tag{6.1.3}$$

Proof By Equation 3.9.1, the definition of the capability matrix is

$$\mathbf{C}_{ij} \equiv \frac{3\sigma_j \cdot \left(\left. \dfrac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{ij}}{(U_i - L_i)/2} \tag{6.1.4}$$

If the pth tolerance width is widened from $(U_p\text{-}L_p)$ to $(U_p\text{-}L_p)(1+\varepsilon)$ then the pth row of the capability matrix becomes

$$\mathbf{C}'_{pj} \equiv \frac{3\sigma_j \cdot \left(\left. \dfrac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{pj}}{(U_p - L_p)(1+\varepsilon)/2} = \left(\frac{1}{1+\varepsilon} \right) \mathbf{C}_{pj} \tag{6.1.5}$$

and the other rows of the matrix are unaffected. By inspection, this is the effect achieved by premultiplying by **T** as defined in 6.1.3.

Theorem 6.1.1 simply states that if a tolerance width is increased, then the corresponding row of the capability matrix and bias vector are reduced and nothing else changes. In some cases, this theorem allows one to make inferences about the effect of tolerance on yield. This may help a system designer decide which tolerances should be relaxed to improve system yield.

Theorem 6.1.2 If the normalized noise vector is composed of independent, normally distributed random variables $\overline{\delta \mathbf{n}}$ with zero mean and the row vectors of $\mathbf{C}$ are mutually orthogonal then

$$\frac{\partial Y_{RT}}{\partial(U_p - L_p)} = \frac{Y_{RT}}{(U_p - L_p)} \frac{3\sqrt{2}}{\sqrt{\pi}} \cdot \frac{1}{\|\mathbf{C}_p\|_2} \cdot \frac{e^{-\frac{9}{2\|\mathbf{C}_p\|_2^2}}}{erf\left(\frac{3\sqrt{2}}{2\|\mathbf{C}_p\|_2}\right)} \tag{6.1.6}$$

Proof Under the conditions prescribed in the theorem, Equation 5.7.1 states that the rolled throughput yield will be

$$Y_{RT} = \prod_{i=1}^{m} \mathrm{erf}\left(\frac{3\sqrt{2}}{2} \frac{1}{\|\mathbf{C}_i\|_2}\right) \tag{6.1.7}$$

From theorem 6.1.1, only the pth row vector will be affected by the change in the pth tolerance width so that

$$\frac{\partial Y_{RT}}{\partial(U_p - L_p)} = \frac{-1}{(U_p - L_p)} \left[Y_{RT} - \prod_{\substack{i=1 \\ i \neq p}}^{m} \mathrm{erf}\left(\frac{3\sqrt{2}}{2} \frac{1}{\|\mathbf{C}_i\|_2}\right) \frac{\partial}{\partial(U_p - L_p)} \mathrm{erf}\left(\frac{3\sqrt{2}}{2} \frac{1}{\|\mathbf{C}_p\|_2}\right) \right] \tag{6.1.8}$$

Applying the chain rule to the derivative in 6.1.8 and collecting terms yields Equation 6.1.6.

Lemma 6.1.1 If the normalized noise vector is composed of independent, normally distributed random variables $\overline{\delta \mathbf{n}}$ with zero mean and the row vectors of $\mathbf{C}$ are mutually orthogonal, then the quality characteristic whose associated row has the largest Euclidean norm has the highest sensitivity to changes in tolerance width.

Proof – The lemma follows directly from theorem 6.1.2. The expression on the right hand side of Equation 6.1.6 is monotonically increasing in the variable $\|\mathbf{C}_i\|_2$ for all positive values of $\|\mathbf{C}_i\|_2$. The value of $\|\mathbf{C}_i\|_2$ is strictly positive.

The theorem above does not apply strictly to the surface mount of MCMs because there is correlation among the quality characteristics. Still it provides a general rule for selecting tolerances to relax. Examining the capability matrix of the system (Equation 4.4.1 is repeated below as 6.1.2), we observe that the Euclidean norms of the leads on sides #2 and #4 are greater than those on sides #1 and #3. If it were possible to widen the tolerance on selected leads (by going to a lower pitch for example), it would be more effective to widen the tolerances on sides #2 and #4 first.

To test this concept, the capability matrix and bias vector of 6.1.2 and 6.1.3 were adjusted to simulate a 50% increase in tolerance width on sides #2 and #4. The rolled throughput yield rose from 66.7% to 89%. By contrast, when the same procedure was repeated for sides #1 and #3, the rolled throughput yield rose from 66.7% to only 75%.

The experiment above demonstrates that, although it applies rigorously in only special circumstances, Theorem 6.1.3 provides a useful general rule for identifying opportunities to increase yield. Relax tolerances on those quality characteristics whose rows have the highest Euclidean norm.

$$\mathbf{C} = \left[0.35 \cdot \mathbf{I}_{368 X 368} \middle| 3.85 \times 10^{-4} \begin{Bmatrix} (i-63)\cdot \mathbf{1}_{125} \\ (i-155)\cdot \mathbf{1}_{59} \\ (i-247)\cdot \mathbf{1}_{125} \\ (i-339)\cdot \mathbf{1}_{59} \end{Bmatrix} \middle| \begin{array}{c|c|c|c} 0.838\cdot \mathbf{1}_{125} & 0.231\cdot \mathbf{1}_{125} & 0.22\cdot \mathbf{1}_{125} & \mathbf{0}_{125} \\ 1.529\cdot \mathbf{1}_{59} & 0.462\cdot \mathbf{1}_{59} & \mathbf{0}_{59} & 0.22\cdot \mathbf{1}_{59} \\ 0.838\cdot \mathbf{1}_{125} & 0.231\cdot \mathbf{1}_{125} & -0.22\cdot \mathbf{1}_{125} & \mathbf{0}_{125} \\ 1.529\cdot \mathbf{1}_{59} & 0.462\cdot \mathbf{1}_{59} & \mathbf{0}_{59} & -0.22\cdot \mathbf{1}_{59} \end{array} \right] \quad (6.1.9)$$

$$\mathbf{k} = 3.85 \times 10^{-3} \begin{Bmatrix} (i-63)\cdot \mathbf{1}_{125} \\ (i-155)\cdot \mathbf{1}_{59} \\ (i-247)\cdot \mathbf{1}_{125} \\ (i-339)\cdot \mathbf{1}_{59} \end{Bmatrix} + \begin{Bmatrix} 0.1\cdot \mathbf{1}_{125} \\ 0.1\cdot \mathbf{1}_{59} \\ -0.1\cdot \mathbf{1}_{125} \\ -0.1\cdot \mathbf{1}_{59} \end{Bmatrix}. \quad (6.1.10)$$

Before leaving the topic of widening tolerances, there is one theorem that can be proven rigorously to hold in all cases regardless of the forms of the distributions and the structure of **C** and **k**.

Theorem 6.1.3 Widening the tolerance on a quality characteristic will either increase rolled throughput yield or leave it unchanged

$$\frac{(U_i - L_i)}{Y_{RT}} \cdot \frac{\partial Y_{RT}}{\partial (U_i - L_i)} \geq 0 \tag{6.1.11}$$

Proof Rolled throughput yield is the integral over the tolerance domain, therefore

$$\partial Y_{RT} = \int_{TD} p_n\left(\overline{\delta \mathbf{n}}\right) d\overline{\delta \mathbf{n}} - \int_{TD'} p_n\left(\overline{\delta \mathbf{n}}\right) d\overline{\delta \mathbf{n}} . \tag{6.1.12}$$

The original tolerance domain is given by

$$TD = \left\{ \overline{\delta \mathbf{n}} \,\middle|\, \begin{bmatrix} \mathbf{C} \\ -\mathbf{C} \end{bmatrix} \cdot \overline{\delta \mathbf{n}} + \begin{Bmatrix} \mathbf{k} \\ -\mathbf{k} \end{Bmatrix} \leq \begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix} \right\} \tag{6.1.13}$$

while the tolerance domain after the relaxation of the pth tolerance is

$$TD' = \left\{ \overline{\delta \mathbf{n}} \,\middle|\, \begin{bmatrix} \mathbf{C}' \\ -\mathbf{C}' \end{bmatrix} \cdot \overline{\delta \mathbf{n}} + \begin{Bmatrix} \mathbf{k}' \\ -\mathbf{k}' \end{Bmatrix} \leq \begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix} \right\} \tag{6.1.14}$$

where **C'** and **k'** are given by Equations 6.1.1 and 6.1.2 respectively. If the tolerance is widened, then ε will be positive and the original tolerance domain contains the relaxed tolerance domain ($TD \subseteq TD'$). Since $p_n\left(\overline{\delta \mathbf{n}}\right)$ is everywhere non-negative, the theorem is proven.

This last theorem should not come as much of a surprise. One would expect that widening product tolerances could never cause the yield of a process to drop. The theorem is included here because other, similarly intuitive notions will prove to be false as demonstrated in Section 6.3.

6.2 Reducing Variances

Manufacturing quality consultants have observed that, in most cases, reducing variation in the top two or three noise factors will significantly improve the process yield. In some circles, this is called "the Pareto principle" [Juran, 1951]. Some call the dominant noise factors "the vital few" [Kalpakjiian, 1993]. Dorian Shainin calls the major contributor "the Red X" [Bhote, 1991]. They also observe that reduction of other causes will have little to no effect on the yield if the

"vital few" noise factors are not addressed. This section will explore this principle by means of the capability matrix approach.

To begin this section, it is essential to determine the effect of a change in the standard deviation of a noise factor on the capability matrix and bias vector.

Theorem 6.2.1 If a system has a capability matrix **C** and bias vector **k** and the qth noise factor has its standard deviation reduced by a fraction ε, then the new capability matrix **C'** and bias vector **k'** will be

$$\mathbf{C}' = \mathbf{CT} \tag{6.2.1}$$

and

$$\mathbf{k}' = \mathbf{k} \tag{6.2.2}$$

where **T** is an n by n matrix such that

$$\mathbf{T}_{ij} = \begin{cases} 0 & \text{if} \quad i \neq j \\ 1-\varepsilon & \text{if} \quad j = p = i \\ 1 & \text{otherwise} \end{cases} \tag{6.2.3}$$

Proof By Equation 3.9.1, the definition of the capability matrix is

$$\mathbf{C}_{ij} \equiv \frac{3\sigma_j \cdot \left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{ij}}{(U_i - L_i)/2} \tag{6.2.4}$$

If the qth standard deviation is reduced from σ_q to $(1-\varepsilon)\sigma_q$ then the qth column of the capability matrix becomes

$$\mathbf{C}'_{iq} \equiv \frac{3(1-\varepsilon)\sigma_q \cdot \left(\left. \frac{\partial \mathbf{q}}{\partial \mathbf{n}} \right|_{\mathbf{n}=\mathbf{t}} \right)_{iq}}{(U_i - L_i)/2} = (1-\varepsilon)\mathbf{C}_{iq} \tag{6.2.5}$$

and the other columns of the matrix are unaffected. By inspection, this is the effect achieved by postmultiplying by **T** as defined in 6.2.3. This proves the first part of the theorem. By the

definition of the bias factor, the Equation 6.2.2 follows directly. This proves the second half of the theorem.

Theorem 6.2.1 simply states that if a noise factor standard deviation is increased, then the corresponding column of the capability matrix is reduced and nothing else changes. Unlike the case of widening tolerances, there do not seem to be any simple closed form expressions for changes in rolled throughput yield with changes in noise standard deviation. However, in general, the noise factors whose corresponding column vector have the highest Euclidean norm tend to be the greatest contributors to defects.

The example of surface mount of MCMs can be used to illustrate the identification of the "vital few." The original capability matrix and bias vector are

$$\mathbf{C} = \left[0.35 \cdot \mathbf{I}_{368 X 368} \middle| 3.85 \times 10^{-4} \left\{ \begin{array}{c} (i-63) \cdot \mathbf{1}_{125} \\ \hline (i-155) \cdot \mathbf{1}_{59} \\ \hline (i-247) \cdot \mathbf{1}_{125} \\ \hline (i-339) \cdot \mathbf{1}_{59} \end{array} \right\} \middle| \begin{array}{c|c|c|c} 0.838 \cdot \mathbf{1}_{125} & 0.231 \cdot \mathbf{1}_{125} & 0.22 \cdot \mathbf{1}_{125} & \mathbf{0}_{125} \\ \hline 1.529 \cdot \mathbf{1}_{59} & 0.462 \cdot \mathbf{1}_{59} & \mathbf{0}_{59} & 0.22 \cdot \mathbf{1}_{59} \\ \hline 0.838 \cdot \mathbf{1}_{125} & 0.231 \cdot \mathbf{1}_{125} & -0.22 \cdot \mathbf{1}_{125} & \mathbf{0}_{125} \\ \hline 1.529 \cdot \mathbf{1}_{59} & 0.462 \cdot \mathbf{1}_{59} & \mathbf{0}_{59} & -0.22 \cdot \mathbf{1}_{59} \end{array} \right] \quad (6.2.6)$$

$$\mathbf{k} = 3.85 \times 10^{-3} \left\{ \begin{array}{c} (i-63) \cdot \mathbf{1}_{125} \\ \hline (i-155) \cdot \mathbf{1}_{59} \\ \hline (i-247) \cdot \mathbf{1}_{125} \\ \hline (i-339) \cdot \mathbf{1}_{59} \end{array} \right\} + \left\{ \begin{array}{c} 0.1 \cdot \mathbf{1}_{125} \\ \hline 0.1 \cdot \mathbf{1}_{59} \\ \hline -0.1 \cdot \mathbf{1}_{125} \\ \hline -0.1 \cdot \mathbf{1}_{59} \end{array} \right\}. \quad (6.2.7)$$

Visual inspection of the capability matrix reveals that noise factor #370 (fourth from the right in the matrix), misalignment of the lead frame on the ceramic body, is likely to have the highest Euclidean norm. An application of the techniques of Section 5 supports this conclusion. If the variance in misalignment were to be reduced to one quarter of its original value, then rolled throughput yield would climb from 66.7% to over 96%. Compare this with an ad hoc strategy. If one were to upgrade the assembly robot to reduce its bias and variance to half its original value, there would be a negligible rise in yield from 66.7% to 67.7%. This contrast in effectiveness of the two strategies demonstrates the value of the Pareto principle in manufacture. It also underscores the value of the capability matrix as a tool for identifying the "vital few".

In practice, it may not be possible to tighten the variance on the misalignment of the lead frame in any direct way. The supplier of the components may have no economically viable

means to improve the capability of his process. Chapter 7 will demonstrate that a similar improvement in yield can be achieved by employing an on-line adjustment strategy.

6.3 Changing the Jacobian

There are a variety of means to reduce the sensitivities of a manufacturing process to noise. Taguchi methods and response surface techniques focus on a good selection of the set points of the process. The basic idea is that at different set points in the feasible operating window, the process may be less sensitive to noise factors.

The sensitivity of a quality characteristic to a noise factor is reflected in the partial derivative $\partial \mathbf{q}/\partial \mathbf{n}|_{\mathbf{n}=\mathbf{t}}$. In general, the response surface of a process over the feasible operating window is nonlinear so that the partial derivative changes with the value of the target point **t**. It would seem that it is desirable, all things being equal, to reduce $\partial \mathbf{q}/\partial \mathbf{n}|_{\mathbf{n}=\mathbf{t}}$. However, for multi-input multi-output problems, there is not just one partial derivative, but an array of partial derivatives (a Jacobian). It is not intuitively obvious which elements are the most critical ones to reduce. In fact, there are cases in which reducing an element of the Jacobian can, against our intuitions, reduce the yield.

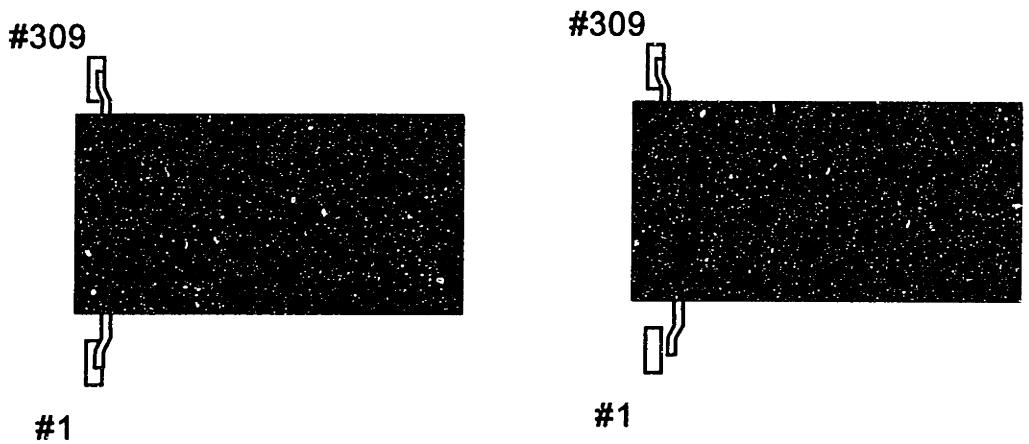
Proposition 6.3.1 There exist process capability matrices for which a reduction in the magnitude of a single element while keeping all the others elements constant will reduce the rolled throughput yield.

Existence Proof If a system has a capability matrix and bias vector of $\mathbf{C} = \begin{bmatrix} 2 & 1 \\ -2 & -1 \end{bmatrix}$ and $\mathbf{k} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$, and its noise factors are independent and normally distributed, then its rolled throughput yield will be 82.2%. If the capability matrix is changed to $\mathbf{C} = \begin{bmatrix} 2 & 0 \\ -2 & -1 \end{bmatrix}$, its rolled throughput yield will be 78.6%..

A physical interpretation of the above capability matrices can be made in the context of the surface mount process. The original matrix corresponds to a process in which we are concerned only with the #1 and #309 leads. Also, let us assume that noise variables #371 and #374 have been amplified by a factor of ten and all the other noise variables have been eliminated. The resulting system will have a capability matrix of $\mathbf{C} = \begin{bmatrix} 2 & 1 \\ -2 & -1 \end{bmatrix}$. Now let us assume the manufacturing system is altered. Instead of placing the lead frame on the component body in one piece, let us say that the lead frame is cut in two. The side containing lead #1 is placed on the component body with near perfect precision so that the upper right element of the matrix is zero. The side containing lead #309 is placed with the same accuracy as the entire lead frame was before. Therefore, the lower right element of **C** remains unchanged. The new capability matrix will be $\mathbf{C} = \begin{bmatrix} 2 & 0 \\ -2 & -1 \end{bmatrix}$. Given that the positioning accuracy of the #1 lead with respect to the MCM body was improved and all other things remained equal, how is it possible that the yield dropped? The reason is that, in some cases, the error in positioning the robot and the error in positioning the frame on the body will cancel to some degree. In the original matrix, this can save a few MCMs from the scrap pile. In the amended case, the cancellation will never save an MCM since only one lead will be corrected and both need to be within tolerance to comply with the acceptance criteria.

MCM with only two leads

$$\mathbf{n}_{371} = 168\mu m$$
$$\mathbf{n}_{374} = 84\mu m$$



$$\mathbf{C} = \begin{bmatrix} 2 & 1 \\ -2 & -1 \end{bmatrix}$$

The noise factor effects partially cancel on both leads.

$$\mathbf{C} = \begin{bmatrix} 2 & 0 \\ -2 & -1 \end{bmatrix}$$

The noise factor effects partially cancel on #309 only.

Figure 6.3.1 An example of inverted sensitivity to noise.

The point of this section is that, for multivariate problems, our intuitions about improving system performance can be poor. Traditional approaches to robust design emphasize reducing sensitivity to noise, but there are cases in which reducing sensitivity to some noise factors will actually degrade the system's performance. It is imperative that, in MIMO systems, robustness should not be equated with low sensitivity to noise. Rather, to define robustness, one should pick a system performance metric and evaluate the system's ability to deliver that performance in the presence of noise.

6.4 Sensitivity of Results to Modeling Error

The previous section developed a formula for change in rolled throughput yield with a change in an element of the capability matrix. This result tells us something about how to change process

set points to improve yield, but it also tells us something about the robustness of the capability matrix approach itself.

The elements of the capability matrix can be considered parameters in a model of a system. The rolled throughput yield is the system performance metric of interest. Some capability matrices will contain hundreds of elements and we can be sure that all of them contain some error. If the rolled throughput yield were sensitive to all of the elements in a large capability matrix, then the chances of getting meaningful results by the capability matrix approach would be very low. However, the results of the last section suggest that only the largest single elements of the matrix will have a significant effect on rolled throughput yield.

6.5 Chapter Conclusions

This chapter has explored one avenue for improving the yield of a manufacturing system. One may balance and tune certain parameters of a system to achieve better performance or greater economy. The three types of system parameters that directly affect the capability matrix and bias vector are the tolerance widths on the quality characteristics, the sensitivities of the quality characteristics to the noise factors, and the variances of the noise factors.

This section showed that widening the tolerance widths never hurts if you can afford to do it. It also proved that, for statistically independent quality characteristics, the most effective tolerance to widen is the one that corresponds to the row with the largest Euclidean norm. For surface mount of MCMs, where there is clearly correlation, the Euclidean norm rule was still effective as a general rule of thumb.

The Euclidean norm rule also seems to work well when applied to column vectors to select key noise factors. In the surface mount of MCMs, the alignment of the lead frame on the component body proved to be the key factor. Reduction of this noise factor was demonstrated to have a strong positive effect on yield.

This chapter also demonstrated the danger of equating robustness and lack of sensitivity to noise. In MIMO systems, a reduction of a single element of the Jacobian may reduce the yield of the system. This underscores the importance of system performance metrics such as rolled throughput yield as the ultimate judge of robustness.

The next chapter introduces another approach to improving system yield. Rather than altering the parameters of the system, this approach changes the architecture of the system by including on-line adjustments.

7. On-line Adjustments

One of the primary uses of manufacturing system block diagrams is in modeling on-line adjustment procedures. Many design for manufacture texts suggest elimination of adjustments in manufacturing processes. Adjustments can lengthen cycle times and may add to quality problems if performed incorrectly [Anderson, 1990]. Yet adjustment processes are still common in manufacture. Many airframe assembly processes involve shimming, all automobiles include adjustments for front end alignment, and many consumer optics products include adjustment screws used in post-assembly calibration. Adjustments are used in industry because, despite their drawbacks, they are often the most economical means to achieving required dimensional accuracy. Our purpose is to provide analytical techniques for assessing the effectiveness of adjustment procedures in manufacture. These techniques may allow engineers to select a small number of adjustments that substantially improve product quality at minimal cost.

7.1 Optimal Multivariate Adjustments

All manufacturing adjustment procedures involve reduction of error in a quality characteristic based on measurement of a disturbance. In other words, adjustment procedures are a form of feedforward control. The mathematical challenge in modeling this type of adjustment stems from the multivariate nature of manufacture. This section presents a means for modeling multivariate manufacturing adjustment procedures as feedforward control schemes applied to manufacturing system block diagrams.

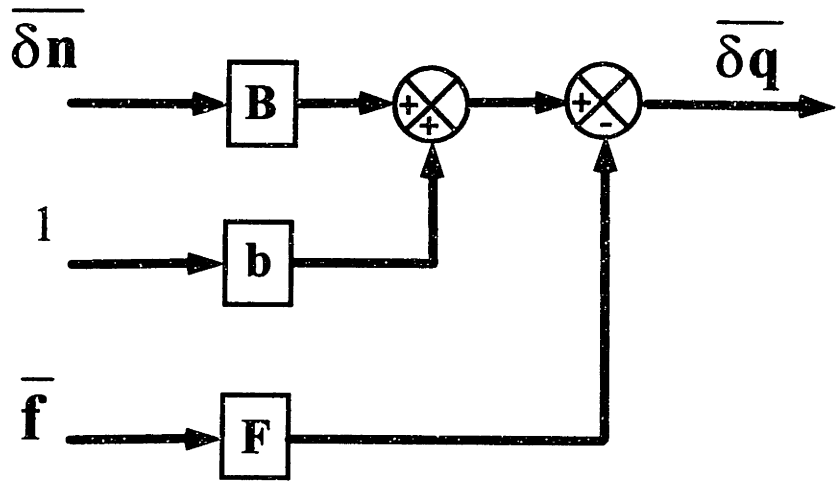


Figure 7.1.1 Block diagram of a manufacturing system with an adjustment variable.

Consider a manufacturing system for which a vector of adjustment variables is available as depicted in Figure 7.1.1. If no adjustment variable were applied, the system would have a process capability matrix and bias vector of **B** and **b** respectively. The input signal $\bar{\mathbf{f}}$ is an f dimensional vector of feedforward adjustment variables. The matrix **F** is the process capability matrix for the adjustment variable. The capability matrix is defined by the Equation 3.9.1 where σ_j and μ_j represent the mean and standard deviation of the error in setting the adjustment variable to the desired value.

The input signal $\bar{\mathbf{f}}$ is available for use to "optimally" reduce the variation in the product as defined by the output signal $\overline{\delta \mathbf{q}}$. The choice of $\bar{\mathbf{f}}$ can be based on a measurement of either the original disturbances, $\overline{\delta \mathbf{n}}$ and 1, or on the quality characteristics, $\overline{\delta \mathbf{q}}$. This thesis will focus on the latter type of adjustment.

To rigorously define what we mean by an "optimal" reduction, we must define an objective function to be minimized. One natural choice is to minimize average quality loss as defined by Taguchi. The average quality loss is defined as

$$Q = \frac{1}{m}\sum_{i=1}^{m} L_i(q_i) \tag{7.1.1}$$

where $L_i(q_i)$ is the quality loss function over the ith quality characteristic [Phadke, 1989]. The quality loss function for a quality characteristic with bi-directional tolerances is defined as

$$L_i(q_i) = \frac{A_o}{\left[(U_i - L_i)/2\right]^2}\left(q_i - \frac{U_i + L_i}{2}\right)^2 \tag{7.1.2}$$

where A_O is the cost of repair or replacement of the product [Phadke, 1989]. Combining Equations 7.1.1 and 7.1.2 with the definition of the normalized quality characteristic vector $\overline{\delta\mathbf{q}}$ Equation 3.11.1, we see that

$$Q = \frac{A_o}{m}\left(\|\delta\mathbf{q}\|_2\right)^2 \tag{7.1.3}$$

where $\|\cdot\|_2$ is the Euclidean norm. Since both A_O and m are constant, the solution that minimizes the Euclidean norm of $\overline{\delta\mathbf{q}}$ will also minimize the average quality loss. Therefore, we will define the optimal adjustment as the one that minimizes $\|\delta\mathbf{q}\|_2$. The problem can be caste as

$$\begin{aligned} &\text{minimize } \left\|\overline{\delta\mathbf{q}} - \mathbf{F}\cdot\bar{\mathbf{f}}\right\|_2 \\ &\text{s.t. } \bar{\mathbf{f}} \in \Re^f \end{aligned} \tag{7.1.4}$$

The problem of Equation 7.1.4 is a linear least squares problem. The solution may be expressed as

$$\bar{\mathbf{f}} = \mathbf{F}^G\left[\overline{\delta\mathbf{q}}\right] \tag{7.1.5}$$

where $\mathbf{F}^G$ is the *generalized inverse* of $\mathbf{F}$ (also known as the Moore-Penrose inverse) [Penrose, 1955]. The generalized inverse of a matrix may be computed by the formula

$$\mathbf{F}^G = \mathbf{V}^T\mathbf{D}^{-1}\mathbf{U} \tag{7.1.6}$$

where $\mathbf{U}^T\mathbf{D}\mathbf{V}$ is the singular value decomposition of $\mathbf{F}$. The generalized inverse will provide a least squares solution for any $\mathbf{F}$. If the solution is not unique, the generalized inverse will provide the least squares solution (or an exact solution if possible) while simultaneously minimizing $\|\bar{\mathbf{f}}\|_2$.

In Figure 7.1.2, the solution of Equation 7.1.5 is used to update the system model of Figure 7.1.1. The noise signal is measured and used to define the optimal adjustment of the vector of adjustment variables. The adjustment is then employed to reduce the final noise of the system so

as to minimize quality loss. The system of Figure 7.1.2 can be reduced to the canonical form as shown in Figure 7.1.3.

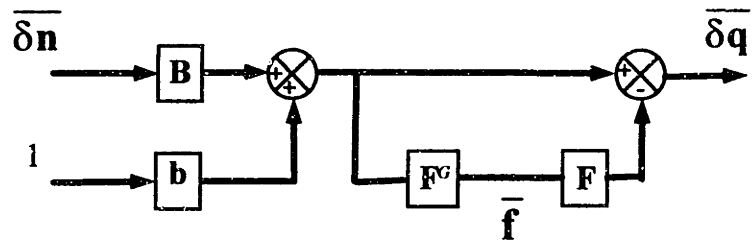


Figure 7.1.2 Block diagram of a manufacturing system with an optimal adjustment.

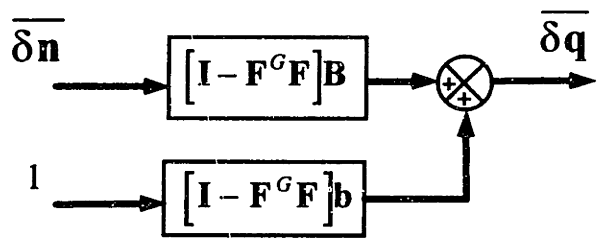


Figure 7.1.3 Reduced form of manufacturing system with an optimal adjustment.

7.2 Errors in Adjustments

The system depicted in Figure 7.1.2 neglects certain practical considerations. In reality there will always be errors due to measurement and actuation. This section will expand the adjustment model of Figure 7.1.2 to include these effects.

To perform an adjustment, one must measure the effect for which the adjustment will compensate. This measurement has limited accuracy and therefore introduces another set of noise factors to the system. These new noise factors require another capability matrix and bias vector (call them **M** and **m**). In the system of Figure 7.2.1, **M** is a diagonal matrix whose elements are the ratio of the standard deviation of the measurement error to the standard deviation of the noise being measured. Similarly, the elements of **m** are the ratios of the mean of the measurement error to the standard deviation of the noise being measured.

Once the desired values of the least squares adjustment have been computed, they must be realized by a physical process. There is likely to be some difference between the desired adjustments and the realized adjustments. This adjustment error can be treated as a random variable and characterized by its mean and standard deviation. These values must be used in defining **F** and **f**, the capability matrix and bias vector of the adjustment variables.

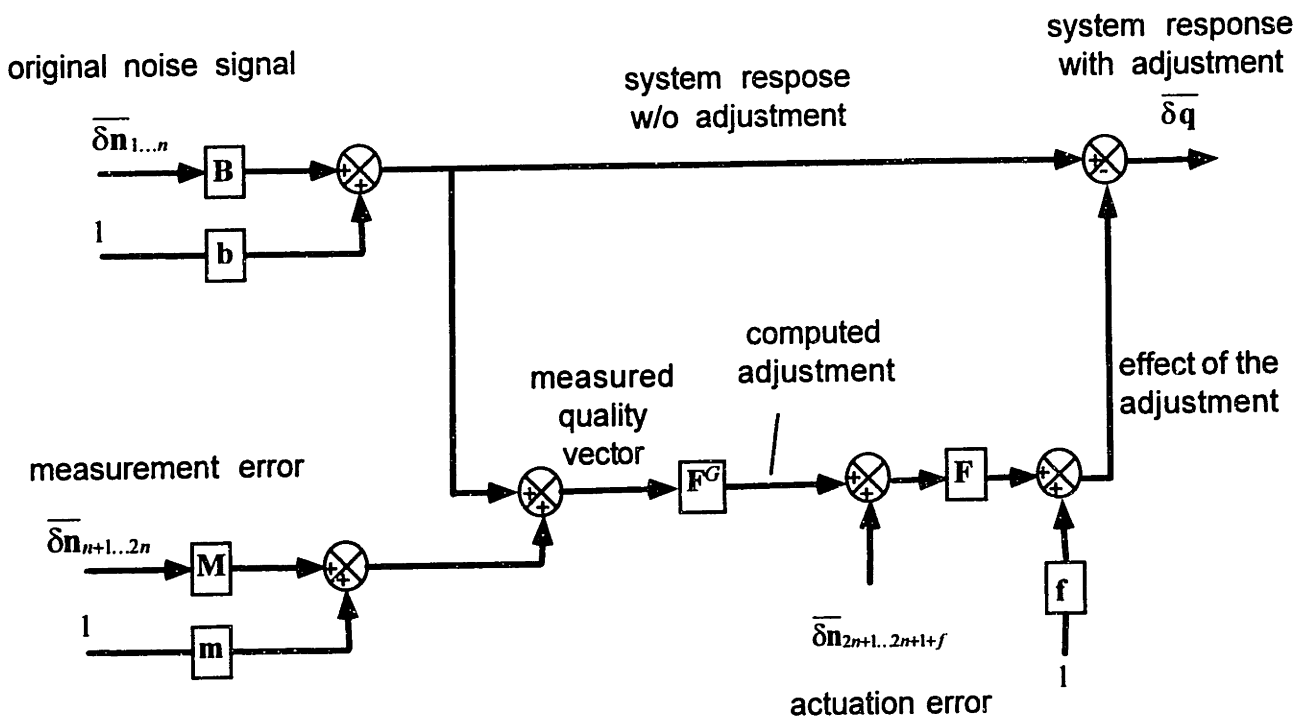


Figure 7.2.1 Block diagram of a manufacturing system with an optimal adjustment including measurement and actuation errors.

The manufacturing system with on-line adjustment control shown in Figure 7.2.1 can be reduced to the equivalent systems of Figure 7.2.2. Note the complexity of the system that is expressed in compact form by a single process capability matrix and bias vector in Figure 7.2.2. These two data structures comprise a complete model of both variation and bias in a system that includes fabrication errors, assembly errors, and on-line adjustment including measurement and actuation error.

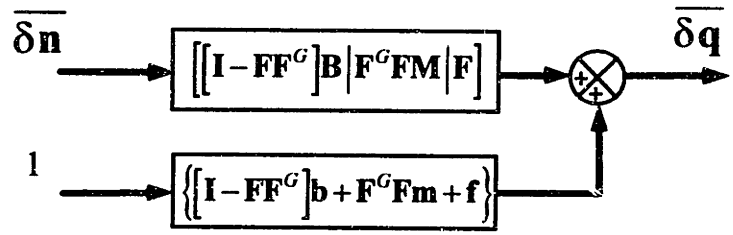


Figure 7.2.2 Reduced block diagram of a manufacturing system with an optimal adjustment including measurement and actuation errors.

As an example of an optimal feedforward adjustment, let us consider the surface mount of MCMs. The robotic assembly system includes a downward looking camera that is used to estimate the centroid of each foot. The robot employs three degrees of freedom (x position, y position, and angular orientation of the MCM) to minimize the sum squared error in side-to-side position of the feet with respect to the pads.

The robotic adjustment procedure can be modeled by a simple reconfiguration of the system model of Figure 4.4.1. The capability matrix for the component placement step may be used as the feedforward capability matrix. A new set of noise factors may be added to account for the error associated with inaccuracy of the visual measurement procedure. We estimated that the measurement process has an error with a standard deviation one tenth that of the signal it is measuring. The new system model is depicted in Figure 7.2.3. The model may be reduced to an overall process capability matrix and bias vector of

$$\mathbf{C}=\left[\begin{array}{c|c|c|c|c|c|c}
0.35\cdot\mathbf{I}_{368X368} & 3.85\times10^{-4}\left\{\begin{array}{c}(i-63)\cdot\mathbf{1}_{125}\\ \hline (i-155)\cdot\mathbf{1}_{59}\\ \hline (i-247)\cdot\mathbf{1}_{125}\\ \hline (i-339)\cdot\mathbf{1}_{59}\end{array}\right\} & \begin{array}{c}0.054\cdot\mathbf{1}_{125}\\ \hline -0.027\cdot\mathbf{1}_{59}\\ \hline 0.054\cdot\mathbf{1}_{125}\\ \hline -0.027\cdot\mathbf{1}_{59}\end{array} & \begin{array}{c}0.231\cdot\mathbf{1}_{125}\\ \hline 0.462\cdot\mathbf{1}_{59}\\ \hline 0.231\cdot\mathbf{1}_{125}\\ \hline 0.462\cdot\mathbf{1}_{59}\end{array} & \begin{array}{c}0.1\cdot\mathbf{1}_{125}\\ \hline \mathbf{0}_{59}\\ \hline -0.1\cdot\mathbf{1}_{125}\\ \hline \mathbf{0}_{59}\end{array} & \begin{array}{c}\mathbf{0}_{125}\\ \hline 0.1\cdot\mathbf{1}_{59}\\ \hline \mathbf{0}_{125}\\ \hline -0.1\cdot\mathbf{1}_{59}\end{array}
\end{array}\right] \tag{7.2.1}$$

$$\mathbf{k}=3.85\times10^{-3}\left\{\begin{array}{c}(i-63)\cdot\mathbf{1}_{125}\\ \hline (i-155)\cdot\mathbf{1}_{59}\\ \hline (i-247)\cdot\mathbf{1}_{125}\\ \hline (i-339)\cdot\mathbf{1}_{59}\end{array}\right\} \tag{7.2.2}$$

The effect of the adjustment process can be surmised by comparing the capability matrix before and after adjustment (Equations 4.4.1 and 7.2.1 respectively). The largest difference is in the fourth column from the right which corresponds to the angular misalignment of the lead frame on the component body. The variation due to this noise factor is almost entirely eliminated by the adjustment process. The only factor that caused the adjustment to be incomplete is the effect of vertical misalignment of the shins as depicted in Figure 2.4.5. The adjustment process has a negligible effect on all the other elements of the matrix. This is supported by intuition. The repositioning of the MCM on the PWA cannot compensate for problems in, for example, scaling of the lead frame.

It is also instructive to compare the bias vectors before and after adjustment (Equations 4.4.2 and 7.2.2 respectively). The bias due to the error in positioning the lead frame on the body is eliminated by the adjustment. Again, the result of the mathematical manipulations supports our intuitions about the system.

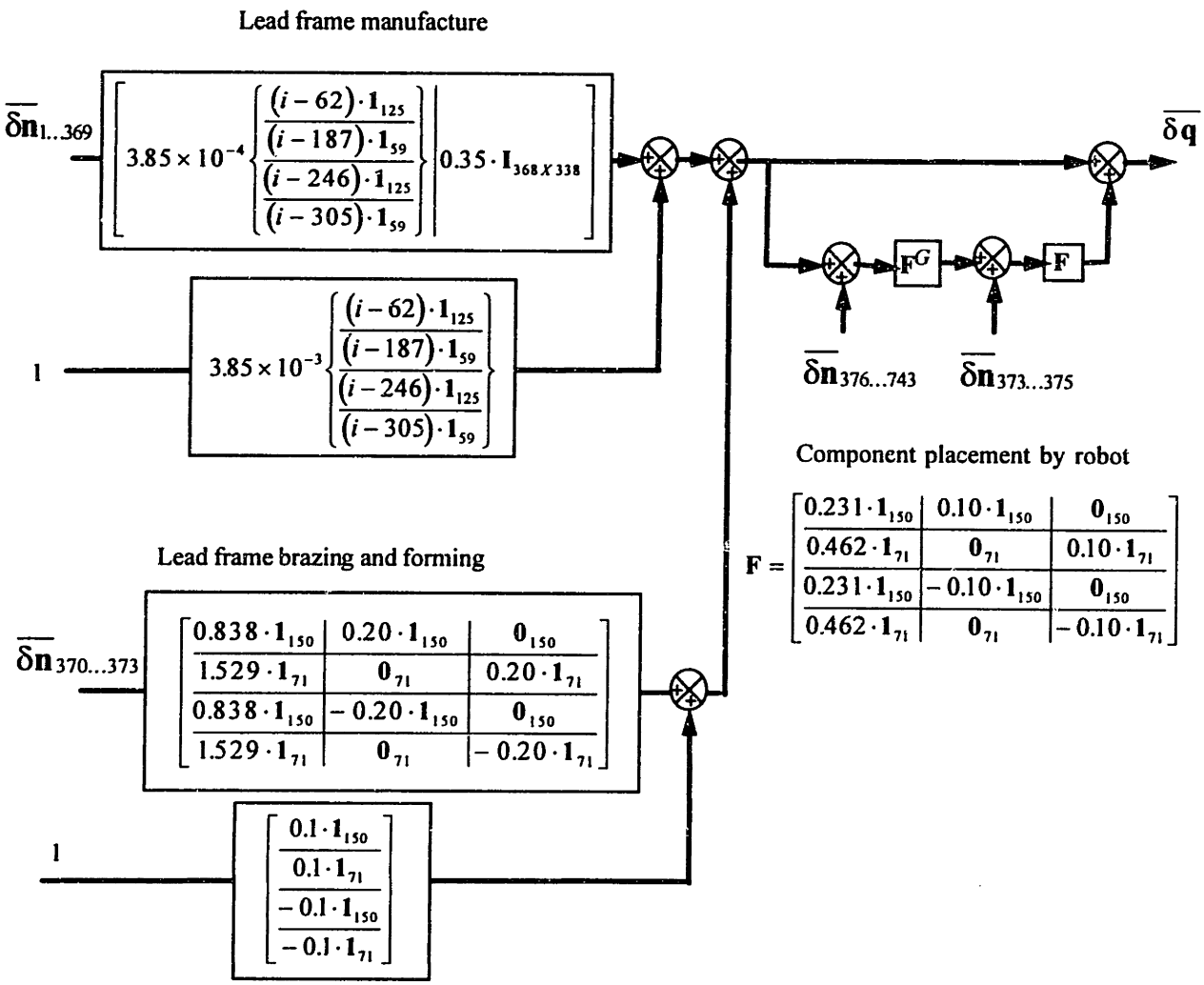


Figure 7.2.3 Expanded block diagram of the electronic assembly process including adjustment.

The block diagram model in Figure 7.2.3 allows one to make predictions about the effect of the adjustment on variance and bias. Nine statistical trials were performed for the surface mount model with optimal adjustments to generate the histogram in Figure 7.2.4. To generate data for comparison, the adjustment procedures were applied to a set of nine 368 leaded MCMs. Histograms of the normalized quality characteristics from the pre-production run are presented in Figure 7.2.4. Comparing Figures 7.2.4 and 7.2.5 reveals that the block diagram adjustment model reasonably predicts the reduction in both bias and variance afforded by the adjustment procedure.

The block diagrams also allow engineers to make inferences concerning system yield. One hundred sets of nine statistical trials generated 92 defect free sets and eight sets with one

defective MCM. Out of the set of nine MCMs manufactured in the pre-production run, one contained defects. This is not the most probable result according to the block diagram model, but the result is not highly improbable either. Much larger data sets would be required to make statistically significant conclusions concerning yield.

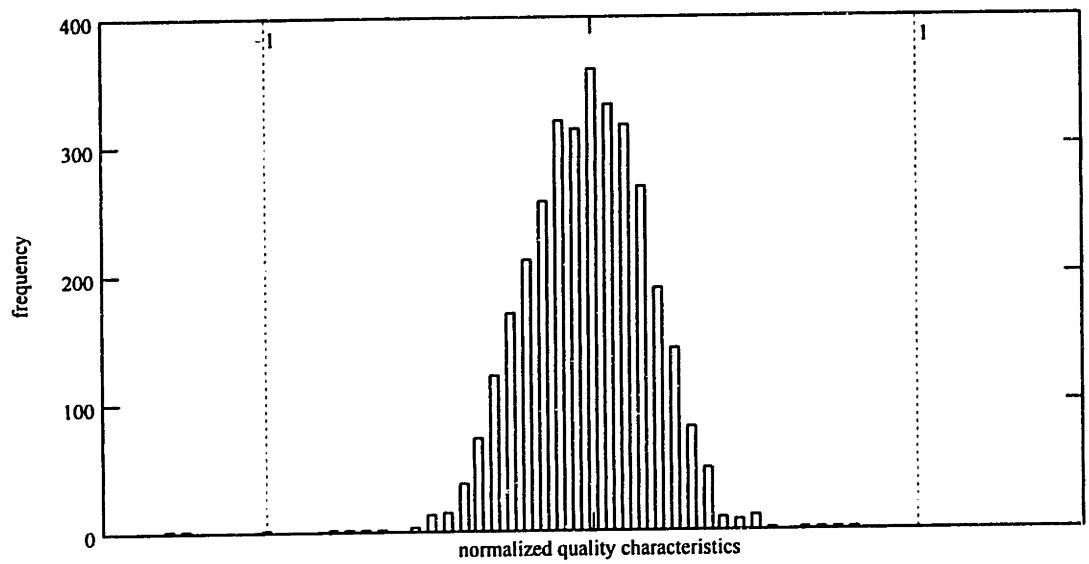


Figure 7.2.4 Data on lead position error from placement of nine 368 leaded MCMs – optimal adjustments applied.

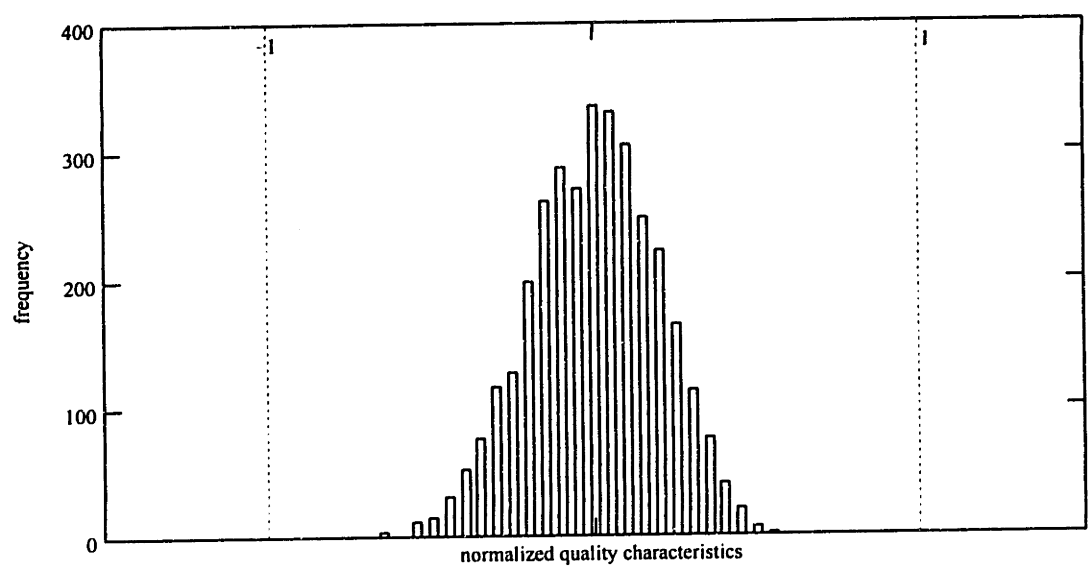


Figure 7.2.5 Simulation results for lead position error from placement of nine 368 leaded MCMs – optimal adjustments applied.

This section developed a model of on-line adjustment procedures and provided an example of their ability to reduce variance and improve yield. The next section concerns selection of minimal sets of adjustment variables required to reap maximum benefits.

7.3 Complete Adjustments

Given that adjustments cost money and take time to perform, one should seek to employ the minimum number of adjustments consistent with meeting one's stated design goals. One goal might be to entirely eliminate quality loss. Although this goal cannot be met in practice, it is instructive to consider the theoretical conditions under which the goal would be attained.

Theorem. The expected value of quality loss for a manufacturing system with an optimal feedforward adjustment and no measurement and actuation error, as represented in Figure 7.1.2, will be zero if and only if $\mathbf{F}$ spans the vector space $\langle \mathbf{B} \rangle$.

Proof. From Penrose [1955], it is a defining property of the generalized inverse that

$$\mathbf{F}\mathbf{F}^G\mathbf{F} = \mathbf{F} \ . \tag{7.3.1}$$

From Ben-Israel [1980], for any matrix $\mathbf{F}$ there exists a full rank factorization so that

$$\mathbf{F} = \mathbf{A}\mathbf{G} \tag{7.3.2}$$

such that $\mathbf{A}$ is a basis for the vector space $\langle \mathbf{F} \rangle$ and $\mathbf{G}$ has full row rank. The generalized inverse may be expressed in terms of the full rank factorization [Ben-Israel, 1980]

$$\mathbf{F}^G = \mathbf{G}^T\left(\mathbf{G}\mathbf{G}^T\right)^{-1}\left(\mathbf{A}\mathbf{A}^T\right)^{-1}\mathbf{A}^T \tag{7.3.3}$$

Combining (7.3.1) and (7.3.2) yields

$$\mathbf{F}\mathbf{F}^G = \mathbf{A}\mathbf{G}\mathbf{G}^T\left(\mathbf{G}\mathbf{G}^T\right)^{-1}\left(\mathbf{A}\mathbf{A}^T\right)^{-1}\mathbf{A}^T \tag{7.3.4}$$

Since $\mathbf{G}$ has full row rank, $\mathbf{G}\mathbf{G}^T$ is non-singular which implies

$$\mathbf{G}\mathbf{G}^T\left(\mathbf{G}\mathbf{G}^T\right)^{-1} = \mathbf{I} \tag{7.3.5}$$

Combining 7.3.4 and 7.3.5 yields

$$\mathbf{F}\mathbf{F}^G = \mathbf{A}\left(\mathbf{A}\mathbf{A}^T\right)^{-1}\mathbf{A}^T \tag{7.3.6}$$

Now **F** spans the vector space $\langle \mathbf{B} \rangle$ iff $\exists \mathbf{A}$ as in 7.3.6 that is a basis for **B**, or

$$\mathbf{B} = \mathbf{AM} \tag{7.3.7}$$

where **M** has full row rank. As above, we may show that

$$\mathbf{BB}^G = \mathbf{A}\left(\mathbf{AA}^T\right)^{-1}\mathbf{A}^T \tag{7.3.8}$$

Combining 7.3.6 and 7.3.8 yields

$$\mathbf{FF}^G = \mathbf{BB}^G \tag{7.3.9}$$

thus

$$\mathbf{FF}^G - \mathbf{BB}^G = 0 \tag{7.3.10}$$

Post-multiplying both sides of 7.3.10 by **B** yields

$$\mathbf{FF}^G\mathbf{B} - \mathbf{BB}^G\mathbf{B} = 0. \tag{7.3.11}$$

From Penrose [1955]

$$\mathbf{BB}^G\mathbf{B} = \mathbf{B} \tag{7.3.12}$$

therefore

$$\mathbf{B} - \mathbf{FF}^G\mathbf{B} = 0. \tag{7.3.13}$$

Rearranging yields

$$\left[\mathbf{I} - \mathbf{FF}^G\right]\mathbf{B} = 0 \tag{7.3.14}$$

The equality in 7.3.14 holds when **B** is replaced with any matrix spanned by **A**. Since **b** is a linear combination of column vectors of **B** (according to Equation 3.10.1) it follows that

$$\left[\mathbf{I} - \mathbf{FF}^G\right]\mathbf{b} = 0 \tag{7.3.15}$$

From Equation 7.1.3

$$E(Q) = \int \frac{A_0}{m}\left(\left\|\mathbf{C}\overline{\delta\mathbf{n}} + \mathbf{k}\right\|_2\right)^2 d\overline{\delta\mathbf{n}} \tag{7.3.16}$$

where A_0 and m are non-zero, and system **C** and **k** are given in Figure 7.1.3. By inspection, *E(Q)* is zero iff the system **C** and **k** are identically zero, which from Figure 7.1.3 is true iff 7.3.14 and 7.3.15 hold.

The theorem proven here has interesting practical consequences for manufacturing system design. Complete disturbance rejection is possible even when both **B** and **F** are coupled (i.e. not triangular). Complete disturbance rejection is also possible when there are fewer adjustment variables than noise factors if **B** does not have full column rank.

7.4 Generalized Inverses and Axiomatic Design

The theorem proven in Section 7.3 can be adapted slightly to address some issues in Axiomatic design. Suh views manufacture as a mapping from a vector of process variables $\{\mathbf{PV}\}$ to a vector of design parameters $\{\mathbf{DP}\}$. This mapping can be represented by a design matrix for the manufacturing process represented as $[\mathbf{B}]$. Thus the mapping is expressed as

$$\{\mathbf{DP}\} = [\mathbf{B}] \cdot \{\mathbf{PV}\} \tag{7.4.1}$$

Suh advances a theorem which states that if $[\mathbf{B}]$ represents a coupled design, then the product cannot be manufactured (Theorem #9 in Suh [1990]). Stated another way, if the mapping between process variables and design parameters is coupled, the process variables cannot in general be set so as to satisfy the design parameters. The theorem proven in Section 7.3 implies, contrary to Theorem #9, that the generalized inverse can be used to set a vector of input variables to match any vector of design parameters even when the mapping is coupled if the matrix $[\mathbf{B}]$ has full row rank. Of course, this assumes that the design matrix $[\mathbf{B}]$ literally represents a linear mapping. To make the above more explicit, we will state it as separate theorem.

Theorem 7.4.1. If $[\mathbf{B}]$ has full row rank, then the linear system $\{\mathbf{DP}\} = [\mathbf{B}] \cdot \{\mathbf{PV}\}$ has an exact solution for all $\{\mathbf{DP}\}$, and the solution is $\{\mathbf{PV}\} = [\mathbf{B}]^G \cdot \{\mathbf{DP}\}$.

Proof. Given a linear system

$$\{\mathbf{DP}\} = [\mathbf{B}] \cdot \{\mathbf{PV}\} \tag{7.4.2}$$

Pre-multiplying by the generalized inverse $[\mathbf{B}]^G$ yields

$$[\mathbf{B}]^G \{\mathbf{DP}\} = [\mathbf{B}]^G [\mathbf{B}] \cdot \{\mathbf{PV}\} \tag{7.4.3}$$

If $[\mathbf{B}]$ has full row rank, then a full rank factorization trivially is

$$[\mathbf{B}] = [\mathbf{I}][\mathbf{B}] \tag{7.4.4}$$

From Ben-Israel [1980], a generalized inverse of [**B**] may be expressed as

$$[\mathbf{B}]^G = [\mathbf{G}]^T\left([\mathbf{G}][\mathbf{G}]^T\right)^{-1}\left([\mathbf{A}][\mathbf{A}]^T\right)^{-1}[\mathbf{A}]^T \tag{7.4.5}$$

where [**B**]=**AG** is a full rank factorization of [**B**]. Combining [**B**]=**AG** with 7.4.4 and 7.4.5 and simplifying yields

$$[\mathbf{B}]^G = \left([\mathbf{B}][\mathbf{B}]^T\right)^{-1}[\mathbf{B}]^T \tag{7.4.6}$$

Post-multiplying by [**B**] yields

$$[\mathbf{B}]^G[\mathbf{B}] = \left([\mathbf{B}][\mathbf{B}]^T\right)^{-1}[\mathbf{B}]^T[\mathbf{B}] \tag{7.4.7}$$

Since [**B**] has full row rank, $[\mathbf{B}][\mathbf{B}]^T$ is non-singular which implies that 7.4.7 reduces to

$$[\mathbf{B}]^G[\mathbf{B}] = [\mathbf{I}] \tag{7.4.8}$$

substituting 7.4.8 into 7.4.3 yields

$$\{\mathbf{PV}\} = [\mathbf{B}]^G\{\mathbf{DP}\} \tag{7.4.9}$$

which proves the theorem.

The conclusion to be drawn from this theorem is that a reasonable process design may be coupled, provided there are at least as many process variables as design parameters and each input affords a linearly independent degree of control.

7.5 Chapter Conclusions

This chapter introduced the use of manufacturing system block diagrams for analyzing multivariate on-line adjustment procedures. The chapter demonstrated that generalized matrix inverses define unique adjustment actions that minimize quality loss as defined by Taguchi. These optimal adjustment procedures can be represented in compact form in a manufacturing system block diagram. Reduction rules allow computation of the system's overall capability matrix and bias vector which is useful in estimating rolled throughput yield. Further, this chapter proved that optimal adjustments based on generalized inverses can be used to entirely reject

disturbances if the appropriate adjustment variables are selected. This chapter also demonstrated that disturbance rejection is possible even in coupled systems.

The surface mount of multi-chip modules served as an example of the value of these adjustment procedure models. Pre-production runs employing least squares adjustments were compared with a block diagram model. The model made reasonable predictions of bias, variance, and yield.

The block diagram modeling techniques introduced here show promise of improving the practice of manufacturing system design. The techniques should allow engineers to more clearly trace the propagation of variation through complex systems. The optimal adjustment model can be used to evaluate proposed system architectures. Given any two proposed adjustment strategies, the methods presented here can be used to evaluate and compare the effectiveness of the two strategies in improving rolled throughput yield.

Even after applying optimal adjustment procedures, there will be a non-zero probability of manufacturing defective units of product. If that probability is too high, then inspection procedures may be used to reduce the probability of shipping defective products. The next chapter introduces the use of the capability matrix as a tool for designing better inspection procedures.

8. Inspection and Key Characteristics

One aspect of manufacturing system design that we have not yet considered is inspection. The purpose of inspection is to reduce the probability that defective products are delivered to the customer. In this sense, inspection serves a valuable function. However, strictly speaking, inspection adds no value to the product. Therefore, it is useful to consider how the risk management function of inspection can be achieved with a minimum number of measurements.

8.1 Redundant Tolerances

Section 5.10 considered the computation of rolled throughput yield for the case of uniformly distributed random variables. The support set S for the noise factors defined the space in which the probability density function over $\overline{\delta\mathbf{n}}$ is non-zero. One way to compute the rolled throughput yield is by integrating the joint density function over the intersection of the support set S and the tolerance domain TD $Y_{RT} = \int_{TD \cap S} p_n\left(\overline{\delta\mathbf{n}}\right) d\overline{\delta\mathbf{n}}$ where

$$TD \cap S = \left\{ \overline{\delta\mathbf{n}} \;\middle|\; \begin{bmatrix} \mathbf{C} \\ -\mathbf{C} \\ \mathbf{I}/\sqrt{3} \\ -\mathbf{I}/\sqrt{3} \end{bmatrix} \cdot \overline{\delta\mathbf{n}} + \begin{Bmatrix} \mathbf{k} \\ -\mathbf{k} \\ 0 \\ 0 \end{Bmatrix} \leq \begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix} \right\} \tag{8.1.1}$$

There is a significant advantage to Lasserre's recursive algorithm for computing the volume of a polytope (see Appendix G). It computes the "volume" of each linear inequality in the set that defines the polytope. If the volume of any inequality is computed to be zero, it constitutes proof that the hyperplane defined by the inequality does not support the polytope $TD \cap S$. Let us call such a linear inequality *redundant*.

The concept of a redundant tolerance is depicted graphically in Figure 8.1.1. In the figure, the constraints on quality characteristics $\overline{\delta\mathbf{q}_1}$ and $\overline{\delta\mathbf{q}_2}$ define a polygon in two dimensions. The constraints on the third quality characteristic do not support that polygon. Therefore the constraints on $\overline{\delta\mathbf{q}_3}$ are said to be redundant.

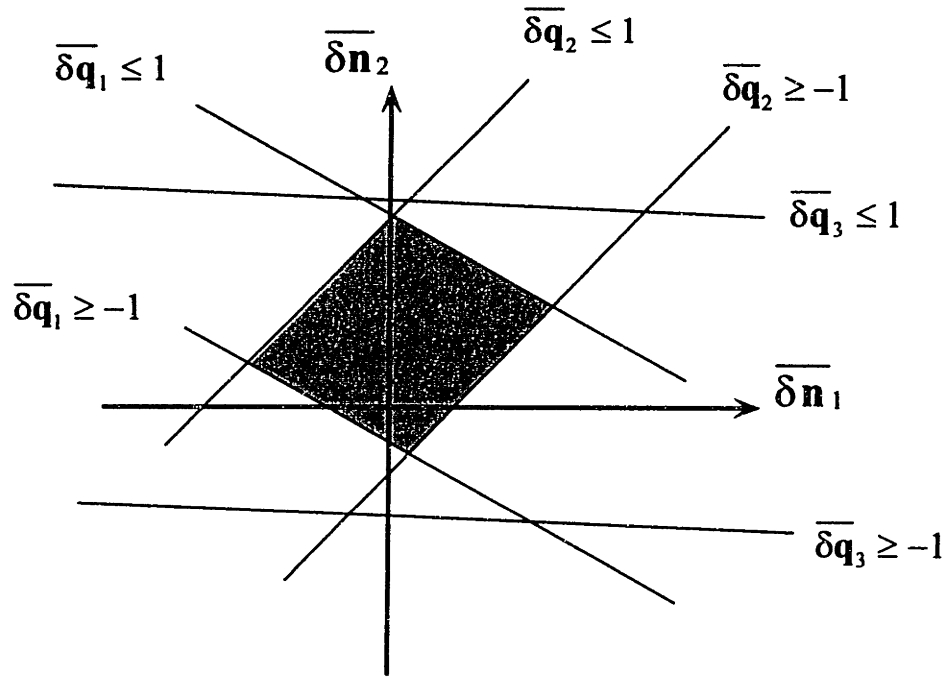


Figure 8.1.1 A redundant tolerance.

Theorem 8.1.1 If a linear inequality is redundant, then one may eliminate that linear inequality from an inspection and still ensure compliance with probability one.

Proof The probability of compliance given that there has been an inspection of all but the ith linear inequality is $\Pr(\overline{\delta \mathbf{n}} \in \{TD \cap S\} | \overline{\delta \mathbf{n}} \in \{TD' \cap S\})$ where TD' is the tolerance domain with the ith linear inequality removed. If the removed inequality is redundant, then it does not support the polytope $TD \cap S$ which implies that $\{TD \cap S\} = \{TD' \cap S\}$. Therefore

$$\Pr(\overline{\delta \mathbf{n}} \in \{TD \cap S\} | \overline{\delta \mathbf{n}} \in \{TD' \cap S\}) = 1. \tag{8.1.2}$$

The theorem above shows that under some conditions, there is nothing to be gained by inspecting certain quality characteristics. The information gained by inspecting other characteristics gives you complete information about the probability of conformance to another characteristic. This result applies only to inspecting for complete certainty. A more general result is proposed in the next section.

8.2 Key Inspection Characteristics

The concept of Key Characteristics (KCs) has been gaining attention as a means for industry to prioritize their efforts. It is recognized that certain dimensions or other measurable characteristics of a product are most important or demand more attention than others. This can be due to the importance of the characteristic to the customer, or because on the incapability of the process to manufacture products that conform to that specification. I propose an alternative conception of KCs especially suited to designing efficient inspection procedures.

Definition A set of quality characteristics is a set of *key inspection characteristics* if and only if the probability of overall conformance is greater than 1-ε if the set of key inspection characteristics conform. The parameter ε is the level of acceptable risk of shipping defective products.

It is desirable to have a test that will determine if a set of quality characteristics is a set of key inspection characteristics. Such a test can be performed by the Monte Carlo method. If the capability matrix is rearranged so that the first 1 through *kpc* quality characteristics are a set of key inspection characteristics, then the Boolean expression below will be true.

$$1-\frac{\sum_{k=1}^{\text{trials}}\left[\prod_{i=1}^{m}\left(\left|\mathbf{C}_i\cdot\text{rand}(n,\ 1/3)+\mathbf{k}_i\right|\le 1\right)\right]}{\sum_{k=1}^{\text{trials}}\left[\prod_{i=1}^{kpc}\left(\left|\mathbf{C}_i\cdot\text{rand}(n,\ 1/3)+\mathbf{k}_i\right|\le 1\right)\right]}<\varepsilon \tag{8.2.1}$$

8.3 Chapter Conclusions

This chapter has briefly explored some of the implications of capability matrices on issues of inspection. The section revealed that some tolerances placed on a product may be redundant. This does not imply that the tolerances are not important, only that if some other set of characteristics are conforming, then, for that production system, there is zero probability that the tolerance will be violated. This idea was extended to allow for some small acceptable probability of non-conformance. This led to a definition of key inspection characteristics and an associated test.

This chapter concludes the theoretical developments related to capability matrices. The next section introduces "virtual machining" – a set of techniques for simulating machining processes. This material is a prerequisite to two case studies – dual head valve grinding and CNC crankpin grinding.

9. Virtual Machining

Before delving into the case studies on machining, it will be necessary to discuss some techniques for machine tool accuracy analysis. Some of these are known in the literature and some were invented and patented by me and my collaborators [Frey, 1997c]. The techniques include approaches to kinematic modeling, computational geometry, and probabilistic simulation. I will call this collection of techniques "virtual machining."

The term "virtual machining" is derived from a report from European Cooperation for Accreditation of Laboratories [1995]. The report suggested the computerization of kinematic models of coordinate measuring machines (CMMs) for error analysis. They suggest that these "virtual CMMs" be used to simulate probing virtual parts to allow more realistic evaluation of CMM capabilities. In this thesis, I apply essentially the same concept to machine tools. I animate a kinematic model and simulate production (rather than probing) of virtual parts.

9.1 Scope and Purpose of Virtual Machining

Virtual machining is computer simulation of machine tool operation. A state of the art simulation might include a detailed finite element analysis of heat flow, dynamic loading, and chip formation. Such detailed simulations can be expensive to develop and are often infeasible (e.g. during conceptual design). Therefore, this chapter focuses on relatively simple models that capture only the kinematic, geometric, and probabilistic behavior of a machine's function. I have found that such models can be developed quickly, are applicable in wide variety of contexts, and provide good accuracy in many cases.

Virtual machining has many practical applications in the design and operation of machine tools. It can be used for:

- Allocating errors to sub-systems and components during design (error budgeting).
- Estimating sensitivity of part tolerances to error sources in the machine tool and its environment.
- Predicting machine tool accuracy in performance tests or in the manufacture of any given part.

- Diagnosing the causes of systematic and random variations in machine performance observed in post-process gauging data.

Although virtual machining can provide significant benefits, it does not appear to be widely used in industry for several reasons. It takes too long to develop customized models for each new machine and each part program. Current modeling techniques impose some important limitations on what manufacturing processes can be efficiently modeled. Many of the potential applications of virtual machining have not been explored.

To reduce the time required to develop machining simulations, I developed an extensible virtual machining framework. The framework handles the problems common to all machine tools (kinematic, geometric, probabilistic, etc.) and allows for more detailed analysis of individual error sources and more sophisticated post-processing of the simulation output as needed for specific applications. The framework is especially suited to implementation in commercial numerical software such as Matlab and Mathcad.

To allow virtual machining to be applied efficiently to grinding and milling I developed an algorithm for mapping error motions into workpiece geometry. The algorithm estimates the location of points on the surface of the work from the location, orientation, direction of travel, and shape of the tool.

To extend the applicability of virtual machining, I developed techniques that allow simulation results to be used for diagnosing machine errors. There exist techniques for manual diagnosis of machine errors based on ball bar test data [Kahino, 1993, Renishaw, 1995]. Virtual machining, by contract, permits diagnosis of errors based on post-process gauging data. I also suggest methods for automated diagnosis by linear regression.

To further extend the applicability of virtual machining, I developed methods that allow tolerances of form to be considered in error budgets. I have developed a program that allows the user to define probability distribution functions for error sources. The program samples at random from the source pdf's to create a population of virtual parts. It then displays the statistical distribution of tolerances of size and form within the population of virtual parts.

The next section begins the discussion of the techniques used to simulate machine operation. The section introduces the kinematic models used to describe the configuration of a machine and its motion.

9.2 Kinematic Models of Machine Tools

In order to simulate machine tool operation, it will be useful to describe the motion of the cutting tool relative to the workpiece. In the virtual machining framework, kinematic models serve this purpose. Kinematic models also serve as an automatic accounting mechanism for the effects of Abbé error – the amplification of small angular errors by large offset distances. This chapter introduces the mathematical formalism and notation used in virtual machining.

9.2.1 Homogeneous Transformation Matrices

There are several methods for constructing kinematic models. Denavit-Hartenberg notation is popular in the robotics literature [Denavit and Hartenberg, 1955]. Some researchers use a formalism based on screw theory [Ziegert, et. al., 1992]. Most researchers in precision machine design and analysis employ homogeneous transformation matrices (HTMs) [Slocum, 1992]. Any of these mathematical notations will be adequate for generating input to the equations derived in this thesis, but I have found it most convenient to use HTMs.

Homogeneous transformation matrices (HTMs) mathematically describe the motion of a rigid body relative to a local coordinate system. A general form of HTM is

$$
{}^{i-1}\mathbf{T}_i = \begin{bmatrix} 1 & 0 & 0 & X \\ 0 & 1 & 0 & Y \\ 0 & 0 & 1 & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.2.1.1)
$$

where X, Y, and Z are displacements in the x, y, and z directions and α, β, and γ are rotations about the x, y, and z axes. This form of HTM allows for large rotations so that local coordinate systems can be re-oriented to model slideways in any orientation and to permit modeling of revolute joints. If more than one of α, β, or γ is non-zero in any given matrix then one should take note of the order in which the rotations are performed. It may make a large difference in the result of the transformation.

The effect of transformation of a rigid body is depicted in Figure 9.2.2.1. Imagine that a block is rotated first about the x axis, then about the y and z axes as depicted on the left hand side of Figure 9.2.2.1. Imagine that this set of rotations is followed by a translation as depicted on the right hand side of Figure 9.2.2.1. If the coordinates of any point on the rigid body before the transformation are (p_x, p_y, p_z), then the coordinates after transformation (p'_x, p'_y, p'_z) are given by

$$\begin{Bmatrix} p'_x \\ p'_y \\ p'_z \\ 1 \end{Bmatrix} = {}^{i-1}\mathbf{T}_i \begin{Bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{Bmatrix} \tag{9.2.1.2}$$

where ${}^{i-1}\mathbf{T}_i$ is given in Equation 9.2.1.1.

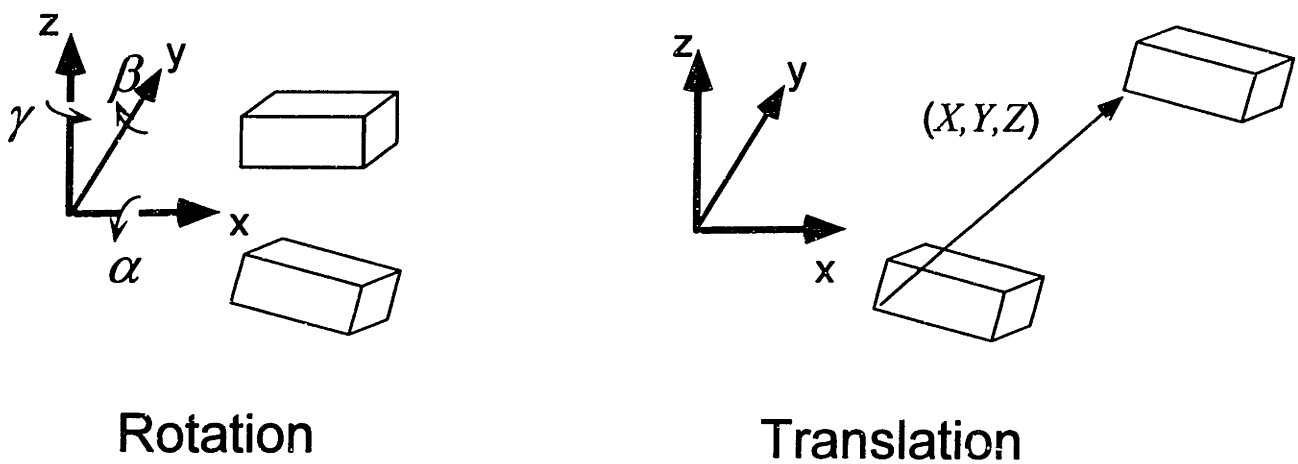


Figure 9.2.2.1 Rotations and translations in a homogeneous transformation matrix.

The details of setting up a kinematic model of a machine tool using HTMs are given in Slocum [1992]. The key concepts and the notation used in virtual machining are reviewed in the next section.

9.2.2 The Tool and Workpiece Chains

A machine tool can be modeled as a set of local coordinate systems (LCS). One LCS is attached to each of the rigid bodies to be represented in the kinematic model of the machine. A reasonable approach is to assign one LCS the base of the machine, one to the workpiece, one to the cutting tool, and one to each carriage. As an example, LCSs are assigned to a surface grinder

in Figure 9.2.2.1. A machine may be further subdivided into individual bearings, actuators, scales, et cetera, as suggested by the requirements of the modeling effort.

The local coordinate systems are usually configured in two serial kinematic chains. One chain extends from the base of the machine out to the tool. The other chain extends from the base out to the workpiece. The base is numbered zero. The coordinate systems in the cutting tool's kinematic chain are number one through *n*. The coordinate systems in the workpiece kinematic chain are number one through *m* (see Figure 9.2.2.2).

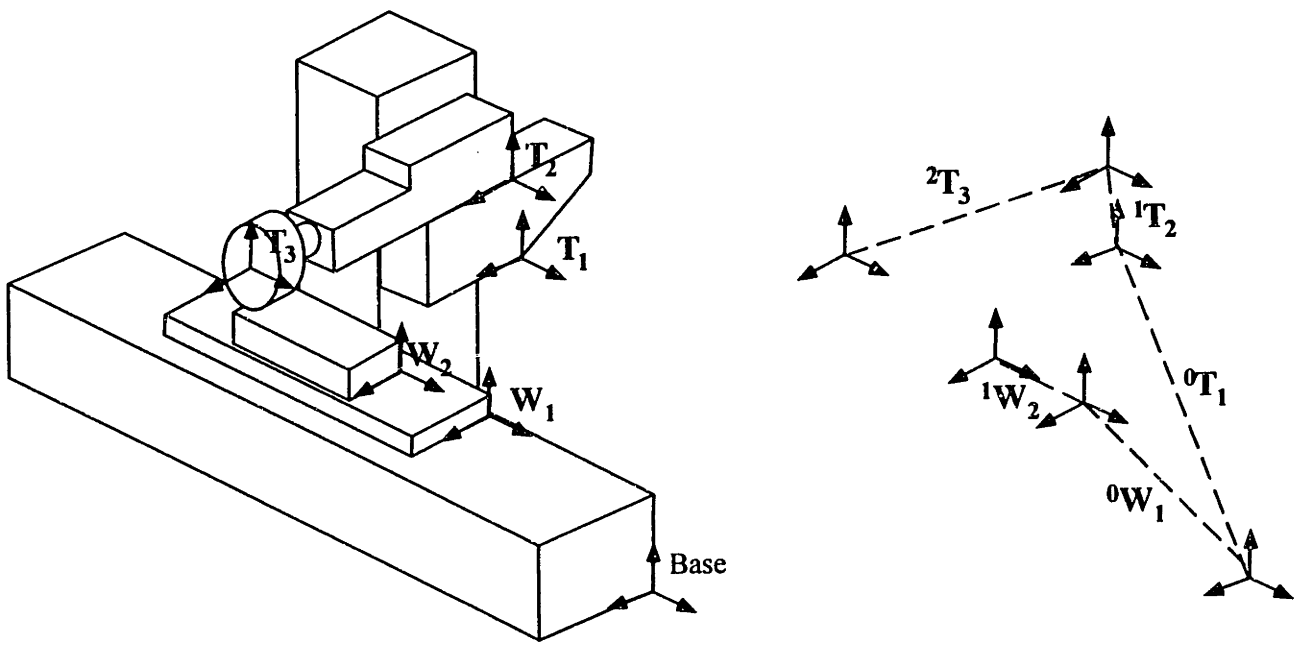


Figure 9.2.2.1 A surface grinder with local coordinate systems and transformation matrices.

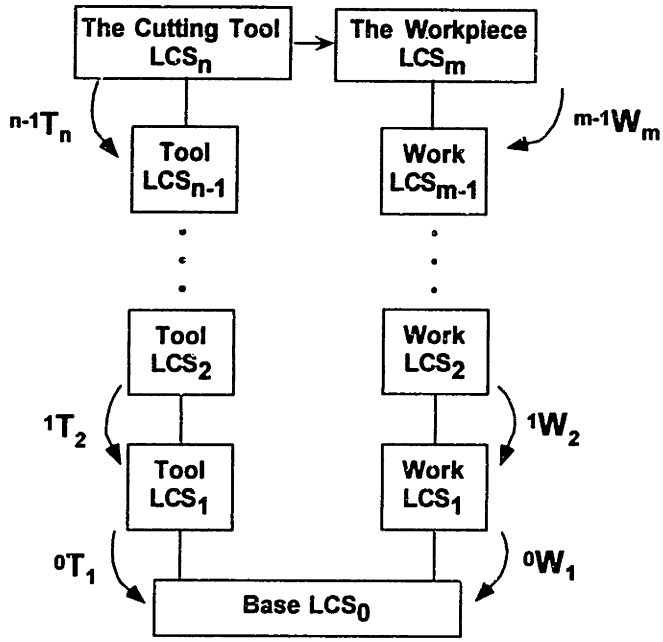


Figure 9.2.2.2 A machine tool as two serial kinematic chains of local coordinate systems.

The coordinates of a point relative to the tool LCS_i can be transformed into coordinates relative to the LCS_{i-1} by multiplying by the HTM ${}^{i-1}\mathbf{T}_i$. Therefore, the HTM that transforms a point on the tool into its equivalent in the base coordinate system is

$$ {}^{0}\mathbf{T}_n = \prod_{i=1}^{n} {}^{i-1}\mathbf{T}_i = {}^{0}\mathbf{T}_1 \cdot {}^{1}\mathbf{T}_2 \cdot \ldots {}^{n-1}\mathbf{T}_n . \tag{9.2.2.1} $$

Similarly, the HTM that transforms a point on the workpiece into its equivalent in the base coordinate system is

$$ {}^{0}\mathrm{W}_m = \prod_{i=1}^{m} {}^{i-1}\mathrm{W}_i = {}^{0}\mathrm{W}_1 \cdot {}^{1}\mathrm{W}_2 \cdot \ldots {}^{m-1}\mathrm{W}_m . \tag{9.2.2.2} $$

Using the methods described so far, one may create an HTM model of a machine at a single position in the machine workspace. The next section shows how to create a model that includes the machine's commanded motions and error motions so that one can model the machine behavior throughout the workspace.

9.2.3 Home, Pose and Error Transformations

Homogeneous transformation matrices can be used to model three distinct aspects of any given machine local coordinate system (LCS): nominal geometry at the home position, commanded motions, and error motions. I will represent the three aspects of each LCS using three separate HTMs: a home transformation $\mathbf{H}_i$, a pose transformation $\mathbf{P}_i(t)$, and an error transformation $\mathbf{E}_i(\mathbf{n})$. Kinematic models with separate pose and error HTMs were first employed by Chao and Yang [1987].

The home position of a machine tool is an arbitrarily chosen position of a machine within its space of possible positions. All motions of the machine are described relative to the home position. For each LCS (excluding the base), one must define an HTM, $\mathbf{H}_i$, which relates its nominal position of the ith LCS to the $(i\text{-}1)^{th}$ LCS in the chain as described in Section 9.2.2.

The pose of a machine tool is the relationship of the machine's commanded position to the home position. The pose of the ith LCS is described by an HTM, $\mathbf{P}_i(t)$. The matrix $\mathbf{P}_i(t)$ transforms the ith LCS from its home position to its commanded position (the positions are measured with respect to the $(i\text{-}1)^{th}$ LCS). Figure 9.2.3.1 depicts a commanded motion of the table on a surface grinder and its representation by an HTM.

To employ the techniques developed in this thesis, the pose HTMs $\mathbf{P}_i(t)$ should be constructed in a spreadsheet or numerical computation package in such a way that they are functions of time. More specifically, the pose matrices should be formulated to reflect the G-code programming for the machining operation to be modeled. If this is accomplished, then the machine will step through the motions required to machine the desired part.

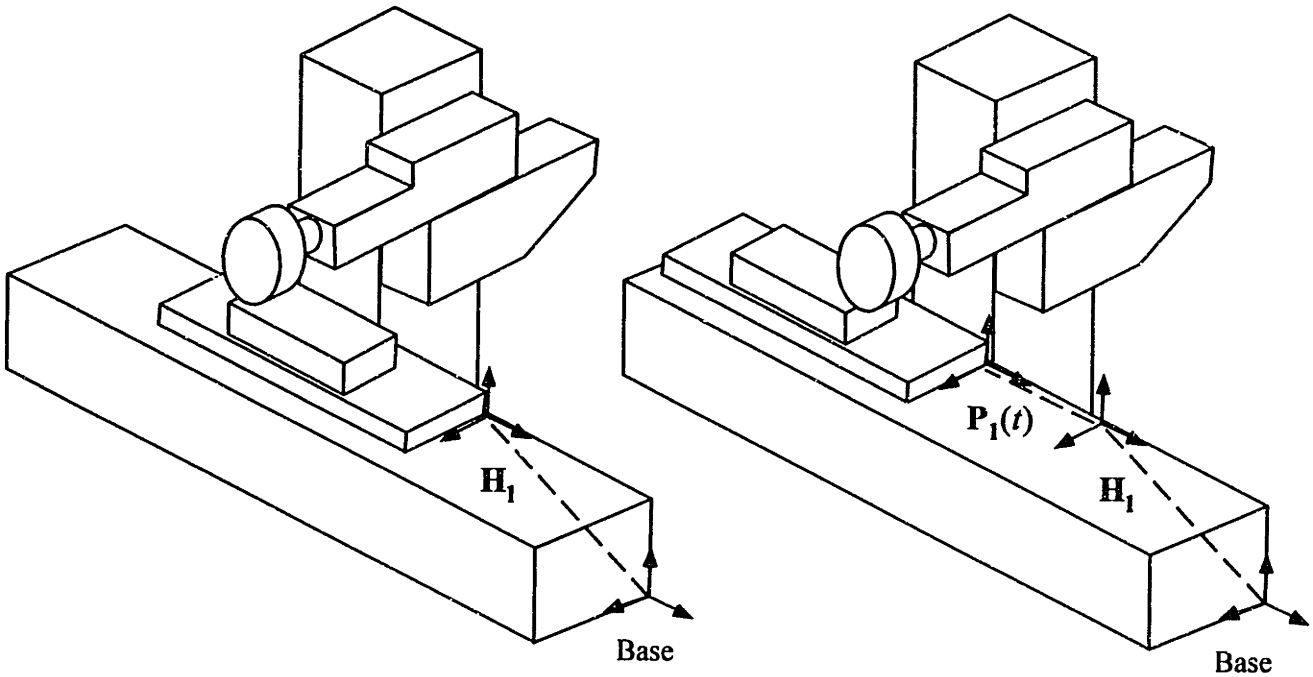


Figure 9.2.3.1 A surface grinder with a pose transformation matrix.

If one creates a model as described so far and includes only home and pose matrices, then one has a model of the nominal operation of the machine for a given task. This thesis is concerned with variation, so this model is clearly incomplete. It is necessary to include the effects of small variations in machine performance in the model. This is accomplished through the error matrices for each LCS.

The error motions of a machine tool are the relationship of the machine's actual position with respect to the commanded position. The error motions of the ith LCS are described by an HTM, $\mathbf{E}_i(\mathbf{n})$ which is a function of the noise vector $\mathbf{n}$ of the manufacturing system. The matrix $\mathbf{E}_i(\mathbf{n})$ transforms the ith LCS from its commanded position to its actual position (the positions are measured with respect to the $(i\text{-}1)^{th}$ LCS). Figure 9.2.2.1 depicts a commanded motion of the table on a surface grinder and its representation by an HTM. The table experiences a pitch about its own LCS axis.

The order of multiplication of the matrices $\mathbf{H}_i$, $\mathbf{P}_i(t)$, and $\mathbf{E}_i(\mathbf{n})$ can have a significant effect on the outcome of the overall transformation. If one wishes to model roll, pitch, and yaw of a typical carriage, it is most appropriate to locate the LCS at the center of stiffness of the carriage (or on the linear scale if the carriage control system uses one) and employ the formula

$$^{i-1}\mathbf{T}_i = \mathbf{H}_i \cdot \mathbf{P}_i(t) \cdot \mathbf{E}_i(\mathbf{n}) \tag{9.2.2.2}$$

or

$$^{i-1}\mathbf{W}_i = \mathbf{H}_i \cdot \mathbf{P}_i(t) \cdot \mathbf{E}_i(\mathbf{n}) \tag{9.2.2.3}$$

depending on whether the LCS is in the tool chain or the workpiece chain.

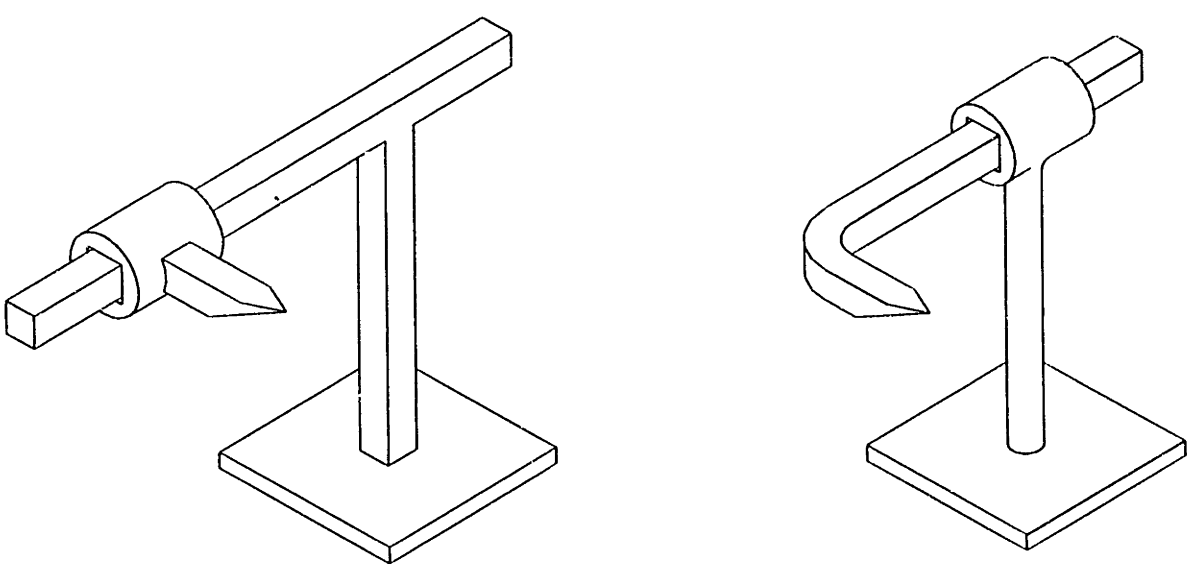
Some machine carriages require a different order of multiplication. If the bearings or bearing surfaces that support a rigid body do not move with that rigid body, then the appropriate formula to use is

$$^{i-1}\mathbf{T}_i = \mathbf{H}_i \cdot \mathbf{E}_i(\mathbf{n}) \cdot \mathbf{P}_i(t) \tag{9.2.2.4}$$

or

$$^{i-1}\mathbf{W}_i = \mathbf{H}_i \cdot \mathbf{E}_i(\mathbf{n}) \cdot \mathbf{P}_i(t) \tag{9.2.2.5}$$

depending on whether the LCS is in the tool chain or the workpiece chain. To illustrate this concept, Figure 9.2.3.2 depicts an example of two means of support for a cutting tool. The one on the left is best modeled by the ordering $\mathbf{H}_i \cdot \mathbf{P}_i(t) \cdot \mathbf{E}_i(\mathbf{n})$. The one on the right is best modeled by the ordering $\mathbf{H}_i \cdot \mathbf{E}_i(\mathbf{n}) \cdot \mathbf{P}_i(t)$.



$\mathbf{T}_1 = \mathbf{H}_1 \cdot \mathbf{P}_1(t) \cdot \mathbf{E}_1(\mathbf{n})$ $\qquad$ $\mathbf{T}_1 = \mathbf{H}_1 \cdot \mathbf{E}_1(\mathbf{n}) \cdot \mathbf{P}_1(t)$

Figure 9.2.3.2 Order of multiplication for the three types of homogeneous transformation.

Figure 9.2.3.3 depicts the effect of correct and incorrect applications of the transformation matrices on the table of a surface grinder. If one wishes to model the yaw of the carriage due to clearance in the bearings, then the true effect of the error motion is depicted on the left of the figure (the error is exaggerated for visual effect). This yaw motion clearly should be performed about the table's LCS *after* the pose is performed; The ordering $\mathbf{H}_i \cdot \mathbf{P}(t)_i \cdot \mathbf{E}_i(\mathbf{n})$ will have the desired effect. If one were to employ the ordering $\mathbf{H}_i \cdot \mathbf{E}_i(\mathbf{n}) \cdot \mathbf{P}_i(t)$ in the kinematic model, then the error motion will be performed about the table's LCS *before* the pose is performed. This will result in an unrealistic Abbé effect in the model as shown in the right hand side of Figure 9.2.3.3. This effect is accounted for by Lin and Ehmann [1993] and by Soons [1993] by defining two different types of transformations for the two types of joints depicted in Figure 9.2.3.2.

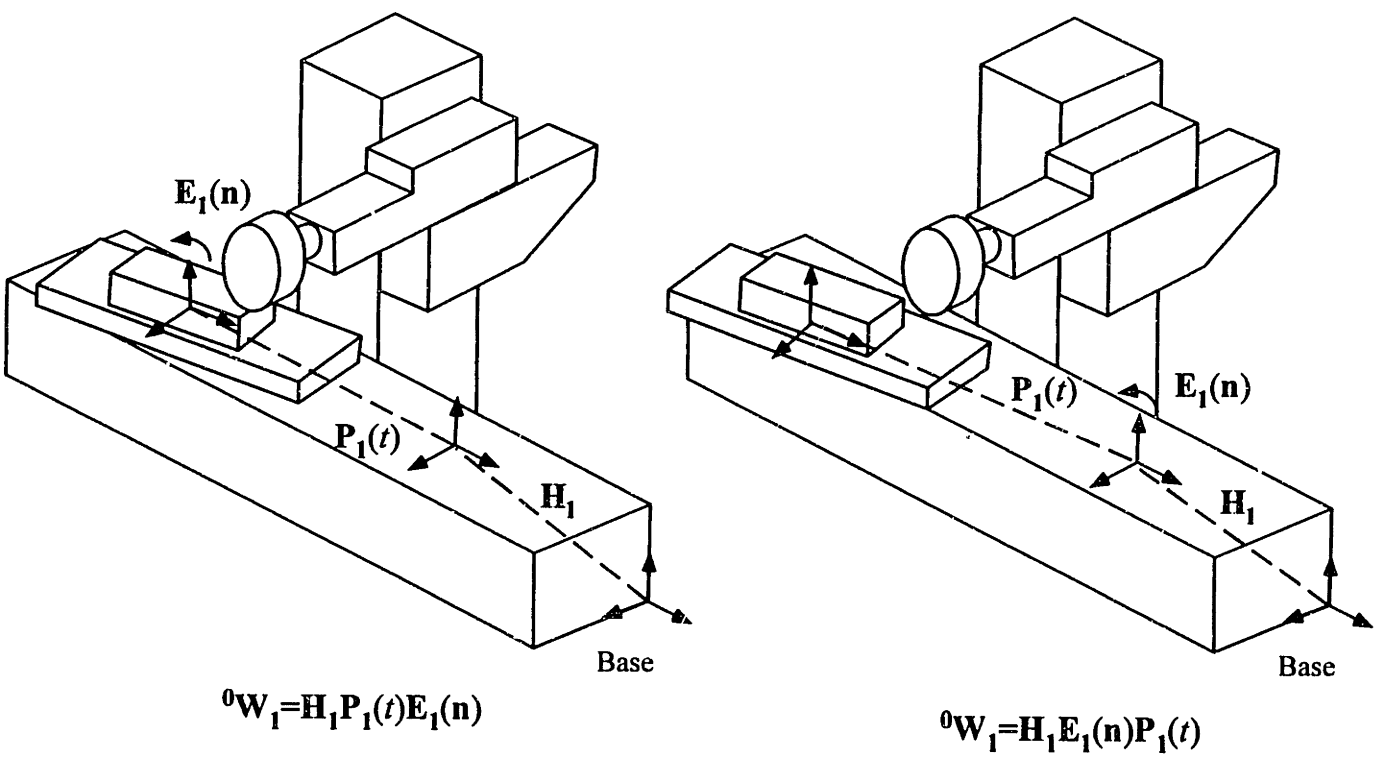


Figure 9.2.3.3 A surface grinder with an error transformation matrix.

An HTM model can be used to compute the location and orientation of the cutting tool with respect to the workpiece at any point in the machining operation. In the following section, I

describe the next important step; I show how to use cutting tool motions to compute workpiece geometry.

9.3 Swept Envelopes of Cutting Tools

One goal of virtual machining is to estimate the accuracy of parts based on a description of a machine tool. The dimensions of interest may be dimensions of size, form, or location. In all three cases, the dimensions can be estimated accurately from a sufficient number of discrete points on the surface the workpiece. Therefore, in this development, I will not attempt to provide methods to develop parametric representations of the surface as is often desired in a CAD system. Instead, I will present methods to calculate discrete points on the surface of the workpiece with the understanding that they will generally be used later to estimate dimensions of the workpiece. Methods of calculating dimensions (e.g. cylindricity, taper, parallelism) from sets of discrete points can be inferred from the definitions in ANSI Y14.5M [1988].

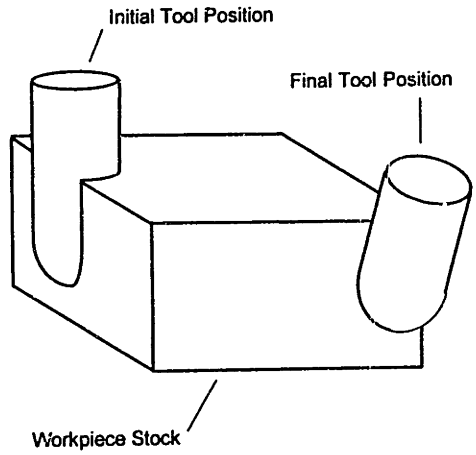
I assume in what follows that a kinematic model of a machine tool exists in the form of equations 9.2.2.1 and 9.2.2.2. This kinematic model represents a simulation of the machine tool in performing the operations required to make a specific part. It contains information on the configuration and dimensions of the machine, the motions required to produce the part, and errors expected to occur in the construction and operation of the machine. The kinematic model can provide information on the relative position, orientation, and motions of the cutting tool and workpiece at any given time t.

9.3.1 Computational Geometry in Machining

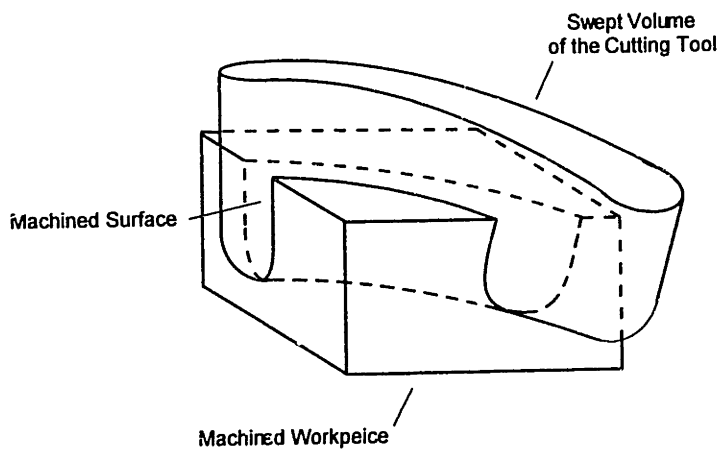
A model of the geometric interaction between the cutting tool and workpiece is an essential component of a machine tool simulation. The mapping from cutting tool positioning error into workpiece geometry error can be formulated as a problem in computational geometry. The problem is to find the envelope surface of the swept volume of a moving solid. A swept volume is the totality of points that belong to the trace of a moving solid over a period of time. The moving solid is known as the generator. The motion function describes the change in position and orientation of the generator as a function of time. The envelope surface is the set of points lying on the surface of the swept volume [Wang and Wang, 1986].

The relevance of the swept envelope problem to machining is illustrated by Figure 9.3.1.1. The cutting tool is the generator. The motion function is determined by the commanded motions and the error motions of the machine tool. The swept volume of the cutting tool is the collection the set of points in the interior of the cutting tool as it moved from its initial position to its final position (Figure 9.3.1.1 a and b). The machined workpiece is the Boolean subtraction from the workpiece stock of its intersection with the swept volume of the cutting tool. For most purposes, it is not necessary to describe the entire machined workpiece mathematically; To evaluate the accuracy of a machine tool one only needs to describe the *machined surfaces* of the workpiece.

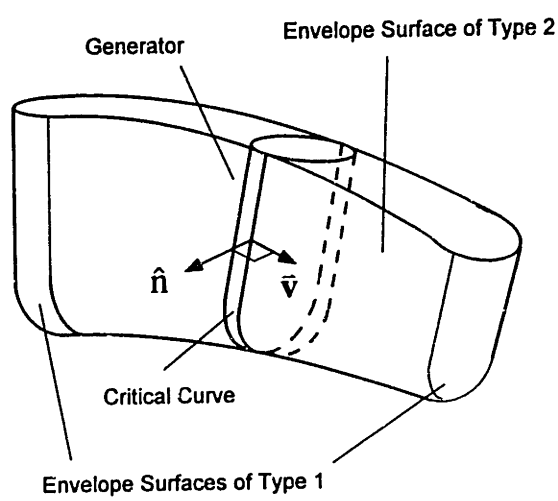
The machined surfaces are the subset of the swept envelope of the cutting tool that lie in the interior of the workpiece stock. The swept envelope of the cutting tool is the set of points on the surface of the swept volume. The swept envelope consists of two types of components: (1) a subset of the boundary of the generator at the initial and final positions and (2) surfaces generated during the motion of the generator [Wang and Wang, 1986] (Fig. 9.3.1.1c). For many machining operations (plunge grinding, horizontal milling, etc.), only the surfaces of type 2 are of interest. In those cases when surfaces of type 1 are of interest, they are not difficult to calculate. Therefore, only surfaces of type 2 will be considered here.



(a) The cutting tool in its initial and final positions with respect to the workpiece



(b) The swept volume of the cutting tool as it moves between the initial and final positions.



(c) The envelope surface of the cutting tool with normal and velocity vectors.

Figure 9.3.1.1 Swept envelopes of cutting tools.

If the computed machined surfaces are based on the *nominal* motions of the cutting tool with respect to the workpiece, then the *nominal machined surface* is computed. A model of nominal machine operation can be created by removing the **E** matrices from the complete kinematic model. If the computation is based on the *perturbed* motions of the cutting tool, then the *perturbed machined surface* is computed. Any information the designer requires concerning the accuracy of the machining process can be estimated by comparing the perturbed and nominal machined surfaces.

Wang and Wang [1986] give a general method for developing an implicit formulation of the envelope surface of a swept volume from a description of a generator curve and a motion function. The key to the method is the realization that, at any point in time, the generator curve is tangent to the envelope surface along a "critical curve" that lies on both the generator and envelope surface. Therefore at any point ($\bar{\mathbf{p}}$) along the tangent curve, the unit normal vector to the envelope surface is identical to the unit normal vector ($\hat{\mathrm{n}}$) of the generator at that point and at that instant in time. Further, the velocity ($\bar{\mathbf{v}}$) of point $\bar{\mathbf{p}}$ must be tangent to the envelope surface. Therefore

$$\bar{\mathrm{v}}(\bar{\mathrm{p}},t)\cdot\hat{\mathrm{n}}(\bar{\mathrm{p}},t)=0\,. \tag{9.3.1.1}$$

Equation 9.3.1.1 above is, in effect, a procedural definition of the envelope surface of a swept solid. In general, finding a point on the envelope surface requires solution of a non-linear equation. If there are no restrictions on the shape of the generator or the motion function, then the solution will require iteration. If one can construct parametric formulations

$$\bar{\mathrm{v}}(\bar{\mathrm{p}},t)=\bar{\mathrm{v}}(u,v,t) \tag{9.3.1.2}$$

$$\text{and } \hat{\mathrm{n}}(\bar{\mathrm{p}},t)=\hat{\mathrm{n}}(u.v,t) \tag{9.3.1.3}$$

then equation 9.3.1.1 will result in a non-linear equation in u, v, an t which all the points on the surface of the envelope must satisfy. Only in special cases will this equation admit reduction to a parametric formulation of the envelope surface.

One can avoid the requirement of explicitly computing the swept envelope by discretizing the problem in time. If one represents the workpiece and cutting tool as solids, then one can

model the machining process as the Boolean subtraction of the cutting tool from the workpiece stock at many points in time (such a system was implemented at General Dynamics for the purpose of NC path verification [Fridshal, 1982]). However, such a calculation requires significant computing resources; The Boolean subtraction must be performed at a large number of discrete times to reduce the size of the resulting "scallops" on the surface to an acceptable size. Considering that most error budgets are implemented in spreadsheets or numerical computation packages, more efficient methods would be of value.

Based on the concept embodied in equation 3, Wang and Wang [1986] developed a package for verification of NC tool paths that simulates the motion of the cutting tool graphically. Their coding scheme is significantly faster than the Boolean subtraction approach. Their result is general except for the restrictions that (1) the motion function must be piece-wise differentiable, (2) the generator must be a convex set, (3) the boundary faces of the generator must be regular surfaces (contain no cusps or self intersections). (Restrictions 2 and 3 are onerous ones for many important machining applications discusssed in Section 9.4.3).

I take the general approach used by Wang and Wang [1986], but enforce some restrictions on the geometry. By considering those special cases of most value in machining operations, I was able to develop closed form equations. This allows for more efficient computation, simpler implementation in packages favored by practicing engineers, and relaxation of some of the restrictions on cutting tool geometry.

9.3.2 Turning, Forming, and Broaching

"Single point" cutting tools, forming tools, and broaches can be modeled by considering only the shape of the cutting edge. The methods developed by Donaldson [1980] and Slocum [1992] work well for such processes. Any point on the cutting edge at any point in time will lie on the surface of the part at any time during the finishing cut (assuming that a positive clearance angle is maintained, see Figure 9.3.2.1). Therefore, the mapping from tool geometry and position to geometry of the part is one to one. Assume a point on the cutting edge of the tool has coordinates (x,y,z) with respect to the cutting tool local coordinate system. Assume that a kinematic model of the machine tool exists in the form of equations 9.2.2.1 and 9.2.2.2. A point,

$\vec{\mathbf{p}}$, on the surface of the workpiece (expressed with respect to the workpiece local coordinate system) cut by point (x,y,z) at time t is

$$\vec{\mathbf{p}}(t)={}_0\mathbf{W}_n^{-1}\cdot{}_0\mathbf{T}_m\cdot\begin{pmatrix}x & y & z & 1\end{pmatrix}^T. \tag{9.3.2.1}$$

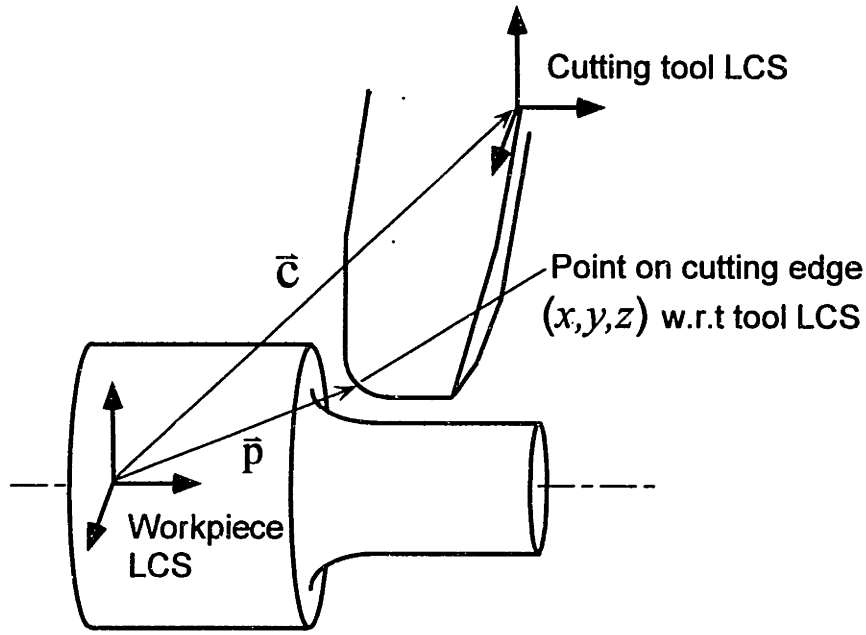


Figure 9.3.2.1 Turning notation.

Spiewak [1987] employed an approach similar to that described above for modeling milling. He included cutting forces in the model and was able to predict the surface finish of milled parts. One difficulty with this approach is that, the majority of the time, any given point on the cutting edge is not producing the final machined surface of the workpiece. Often, the point is creating a machined surface that is itself machined away a short time later by another cutting edge. In other cases, the point is not in contact with the workpiece. Therefore, significant post processing is required to find the intersections of the multiple surfaces calculated by the model. Where one is only concerned with the *form* of parts rather than *surface finish* in milling operations, it is more convenient to use the approach described in the next section.

9.3.3 Form Grinding and Milling

Cutting tools that generate the surface speed of the cutting edge by rotating about their axis of symmetry (such as grinding wheels and milling cutters) can, under certain conditions, be modeled as surfaces of revolution. The modeling assumptions are:

- The z axis of the tool local coordinate system lies on the cutting tool's axis of rotation.
- The shape of the cutting tool is defined by a function $r(u)$, equal to the perpendicular distance of the cutting edge (or grain) from the axis of rotation as a function of u, the component along the axis of rotation of the distance from the cutting point to the origin of the tool local coordinate system (Fig. 9.3.3.1).
- Points on the envelope surface swept out by the cutting tool are also a points on the surface of the workpiece. In some cases only a subset of the points on the envelope surface will actually be on the surface of the part (for example, when the swept envelope intersects itself).

The assumptions listed above imply that I neglect such effects as built up edge and excessive feed rates leading to scalloping in the case of milling cutters.

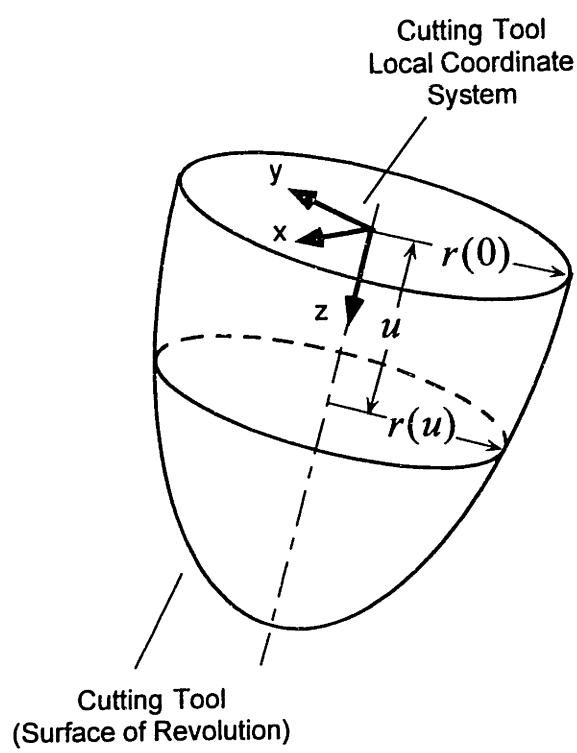


Figure 9.3.3.1 Representing a cutting tool as a surface of revolution.

Given the above assumptions, the derivation of the closed form equations for points on the envelope surface is a natural extension of the method for cylinders given by Wang and Wang [1986]. I follow their notation where appropriate and combine it with the notation of the kinematic model given above. A graphical representation of the variables introduced in equations 9.3.3.1 through 9.3.3.14 below is provided in Figure 9.3.3.2.

The location of the origin of the tool LCS with respect to the workpiece is

$$\bar{\mathbf{c}}(t)={}_0\mathbf{W}_n^{-1}\cdot{}_0\mathbf{T}_m\cdot\begin{pmatrix}0 & 0 & 0 & 1\end{pmatrix}^T. \tag{9.3.3.1}$$

The unit vector pointing along the axis of rotation of the tool is

$$\hat{\mathbf{a}}(t)={}_0\mathbf{W}_n^{-1}\cdot{}_0\mathbf{T}_m\cdot\begin{pmatrix}0 & 0 & 1 & 0\end{pmatrix}^T. \tag{9.3.3.2}$$

Using $\hat{a}$ and $\bar{c}$, one may define a local coordinate system on the tool defined by the set of mutually orthogonal unit vectors $\hat{e}_1$, $\hat{e}_2$, and $\hat{e}_3$. The local coordinate system is defined as

$$\hat{e}_1 = \hat{a}$$

$$\text{if} \quad |\dot{\hat{a}}| \neq 0 \quad \text{then} \quad \hat{e}_2 = \frac{\dot{\hat{a}}}{|\dot{\hat{a}}|} \quad \text{and} \quad \hat{e}_3 = \hat{e}_1 \times \hat{e}_2 \tag{9.3.3.3}$$

$$\text{if} \quad |\dot{\hat{a}}| = 0 \text{ and } \dot{\hat{a}}\cdot\dot{\bar{c}} \neq 0 \quad \text{then} \quad \hat{e}_3 = \frac{\hat{a}\times\dot{\bar{c}}}{|\hat{a}\times\dot{\bar{c}}|} \quad \text{and} \quad \hat{e}_2 = \hat{e}_3 \times \hat{e}_1$$

This definition allows for consideration of machining operations in which the cutting tool's position *or* orientation are fixed at some instant in time, but not operations in which both position *and* orientation are simultaneously fixed at some instant in time. If the tool dwells, then every point on the surface of the cutting tool must be considered a potentially valid surface point on the workpiece. If the tool is plunging along its axis of revolution then the cutting tool creates a cylinder with a radius equal to the maximum radius of the tool. This is a special case that will not be considered further in this thesis.

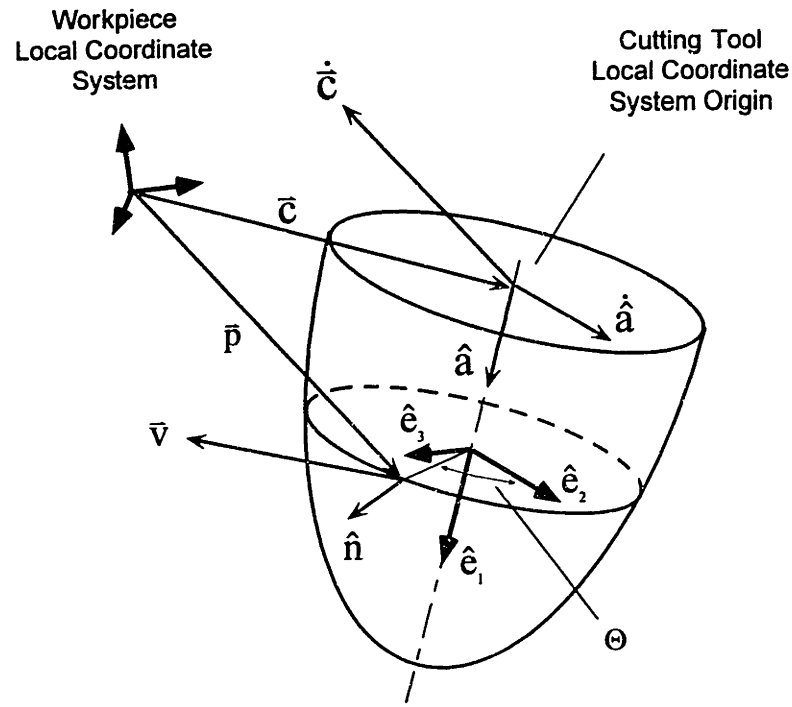


Figure 9.3.3.2 Notation for calculating points on the critical curve.

Consider the circle defined by the intersection of a plane normal to the cutting tool's axis of rotation, $\hat{a}$, offset from the origin of the tool coordinate system, $\bar{c}$, by the perpendicular distance u. A point $\bar{p}$ lies in that plane at an angle θ measured from the axis $\hat{e}_2$. The velocity of point is

$$\bar{v} = \dot{\bar{c}} + u|\dot{\hat{a}}|\hat{e}_2 - r(u)|\dot{\hat{a}}|\hat{e}_1 \cos\theta \tag{9.3.3.4}$$

The unit normal vector to the surface of revolution at point $\bar{p}$ is

$$\hat{n} = \frac{-r'(u)\hat{e}_1 + \hat{e}_2 \cos\theta + \hat{e}_3 \sin\theta}{\sqrt{1 + r'(u)^2}} \tag{9.3.3.5}$$

where

$$r'(u) = \left.\frac{dr}{dz}\right|_{z=u} . \tag{9.3.3.6}$$

Combining equations 9.3.3.4 and 9.3.3.5 into equation 9.3.1.2.1 yields an equation of the form

$$A\cos\Theta + B\sin\Theta + C = 0 \tag{9.3.3.7}$$

where

$$A = |\dot{\bar{a}}|r'(u)r(u) + \dot{\bar{c}}\cdot\hat{e}_2 + |\dot{\bar{a}}|u \tag{9.3.3.8}$$

$$B = \dot{\bar{c}}\cdot\hat{e}_3 \tag{9.3.3.9}$$

$$C = -r'(u)\dot{\bar{c}}\cdot\hat{e}_1 \tag{9.3.3.10}$$

Equation 9.3.3.7 admits a closed form solution

$$D = \sqrt{B^2 + A^2 - C^2} \tag{9.3.3.11}$$

$$\cos\theta = \frac{C^2 - AC - B^2 + BD}{A^2 - AC + B^2 - BD} \tag{9.3.3.12}$$

$$\sin\theta = \frac{(A-C)(B-D)}{A^2 - AC + B^2 - BD}. \tag{9.3.3.13}$$

The results of equations 9.3.3.12 and 9.3.3.13 can be used to compute the positions of the points $\bar{p}(t,u)$ on the surface of the working envelope

$$\bar{p}(t,u) = \bar{c} + u\hat{e}_1 + r(u)(\hat{e}_2\cos\theta \pm \hat{e}_3\sin\theta). \tag{9.3.3.14}$$

The ± symbol indicates that there are two solutions for $\bar{p}(t,u)$. If the tool is cutting a trough, then both solutions are valid. If the tool is contouring counter-clockwise as viewed in the positive $\hat{e}_3$ direction, then retain only the solution corresponding to the plus sign. If the tool is contouring clockwise as viewed in the positive $\hat{e}_3$ direction, then retain only the solution corresponding to the minus sign.

There are important conditions on the equations above that should be checked. The angular separation of the two points on the critical curve that lie in a plane offset by a distance u from the origin is 2θ (see Figure 9.3.3.3). If θ equals zero, then the critical curve is tangent to the

circle formed by the intersection of a plane normal to the axis of rotation. As is evident from equation 9.3.3.7 above, if A is equal to -C then $\sin\theta$ is zero, $\cos\theta$ is one, and θ equals zero. For any values of u such that A-C<0, that cross section of the tool is not in contact with the workpiece surface and the results of the equation 9.3.3.14 above must be discarded. Similarly, if the value of D is imaginary, the results of the calculation can be safely disregarded. This occurs only when the tool is plunging at a steep angle so that every point cut by that section of the tool is immediately swept away by the cross sections following behind it. These special cases will not normally be encountered in modeling form grinding or horizontal milling applications.

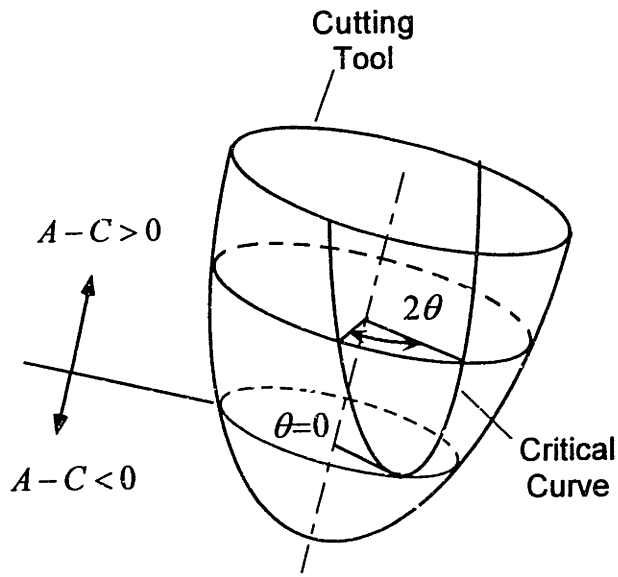


Figure 9.3.3.2 The physical meaning of vanishing theta.

9.3.4 Ball End Milling

Ball ended milling cutters (used frequently for milling sculpted surfaces) fit the criteria of case 2 above, but their spherical shape permits simpler analysis. If one assumes that only the spherical portion of the milling cutter is in contact with the workpiece on any of the finishing cuts (as will normally be the case), then the swept envelope will be a portion of a canal surface. The methods of computing canal surfaces from their spine curves were first described by Gaspard Monge [1851]. The critical curve of a canal surface is always a circle with the same radius of the swept sphere and always lies in a plane perpendicular to the spine curve. Therefore, the method below for calculating points on the surface of the workpiece follows.

Place the origin of the cutting tool's local coordinate system at the center of the spherical portion of the cutter. The coordinates of the center of the sphere $\bar{c}$ are

$$\bar{c}(t) = {}_0\mathbf{W}_n^{-1} \cdot {}_0\mathbf{T}_m \cdot \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix}^T. \tag{9.3.4.1}$$

Define the geometry of the tool so that the z axis is the axis of rotation. The tool's z axis mapped into the workpiece coordinate system is therefore

$$\hat{a} = {}_0\mathbf{W}_n^{-1} \cdot {}_0\mathbf{T}_m \cdot \begin{pmatrix} 0 & 0 & 1 & 0 \end{pmatrix}^T. \tag{9.3.4.2}$$

A point on the surface of the workpiece is

$$\bar{p}(t,\theta) = \bar{c} + r\hat{e}_2 \cos\theta \pm r\hat{e}_3 \sin\theta \tag{9.3.4.3}$$

$$\text{where} \quad \hat{e}_3 = \frac{\hat{a} \times \dot{\bar{c}}}{|\hat{a} \times \dot{\bar{c}}|} \tag{9.3.4.4}$$

$$\text{and} \quad \hat{e}_2 = \hat{e}_3 \times \hat{a} \tag{9.3.4.5}$$

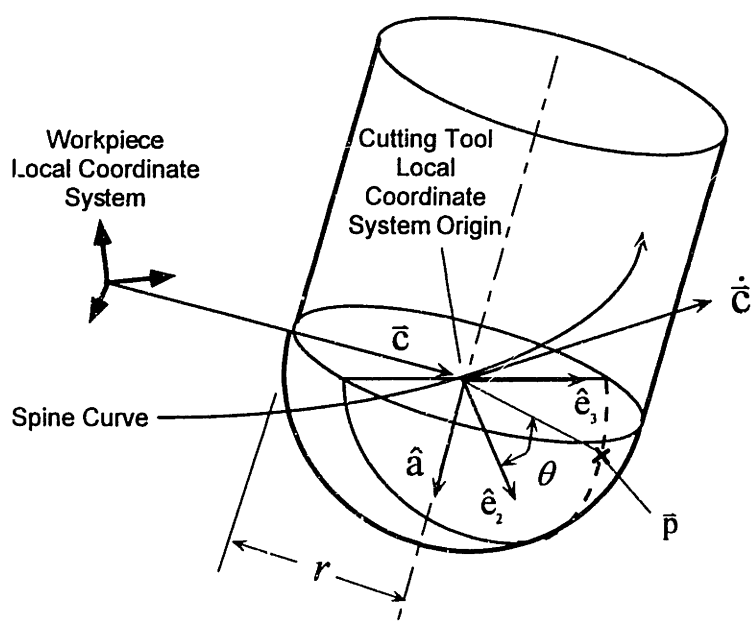


Figure 9.3.4.1 Notation for calculating the swept envelope of a ball ended milling cutter.

The parameter θ may range from $-\pi$ to $+\pi$ at most. In general the valid range is determined by the spacing of the machining passes. Refer to Vickers and Quan [1985] for a discussion of scallop heights versus machine pass spacing.

This section has described methods to compute points on the swept envelope of cutting tools. The next section describes a way to combine the swept volume algorithms and the HTM modeling method to simulate machining operations.

9.4 Running Simulations

9.4.1 Simulation Structure

Sections 9.2 and 9.3 provide the mathematical tools needed to create a machining simulation. This section describes the structure of a simulation that employs these tools. This structure is depicted in Figure 9.4.1.1.

There are four main inputs required for a machining simulation: nominal geometry, commanded motions, a noise vector, and cutting tool shape. The nominal geometry of the machine at the home position may be derived from engineering drawings of the machine tool or from other appropriate sources such as a CAD data base. The nominal geometry determines the home matrices $\mathbf{H}_i$. The commanded motions of the machine in manufacturing a given part may be derived from mathematical models of the controller or from G-Codes. The commanded motions determine the pose matrices $\mathbf{P}_i(t)$. The noise vector is determined by positing noise factors and assembling them into a vector as described in Sections 3.4 and 3.5. The noise vector determines the error matrices $\mathbf{E}_i(\mathbf{n})$. The cutting tool shape may be a given or it may be computed based on a prior simulation of a dressing process (as will be demonstrated in Chapter 10).

The HTM matrices for a machine, when multiplied together in the proper order (see Sections 9.2.2.and 9.2.3), constitute a kinematic model of a machine tool. The HTM model can be used to compute, for any time t, the cutting tool's position, velocity, orientation, and change in orientation with time. These variables may be combined, for any axial position u, with the local radius and slope of the cutting tool to determine one or two points $\bar{\mathbf{p}}(t,u)$ on the surface of the workpiece.

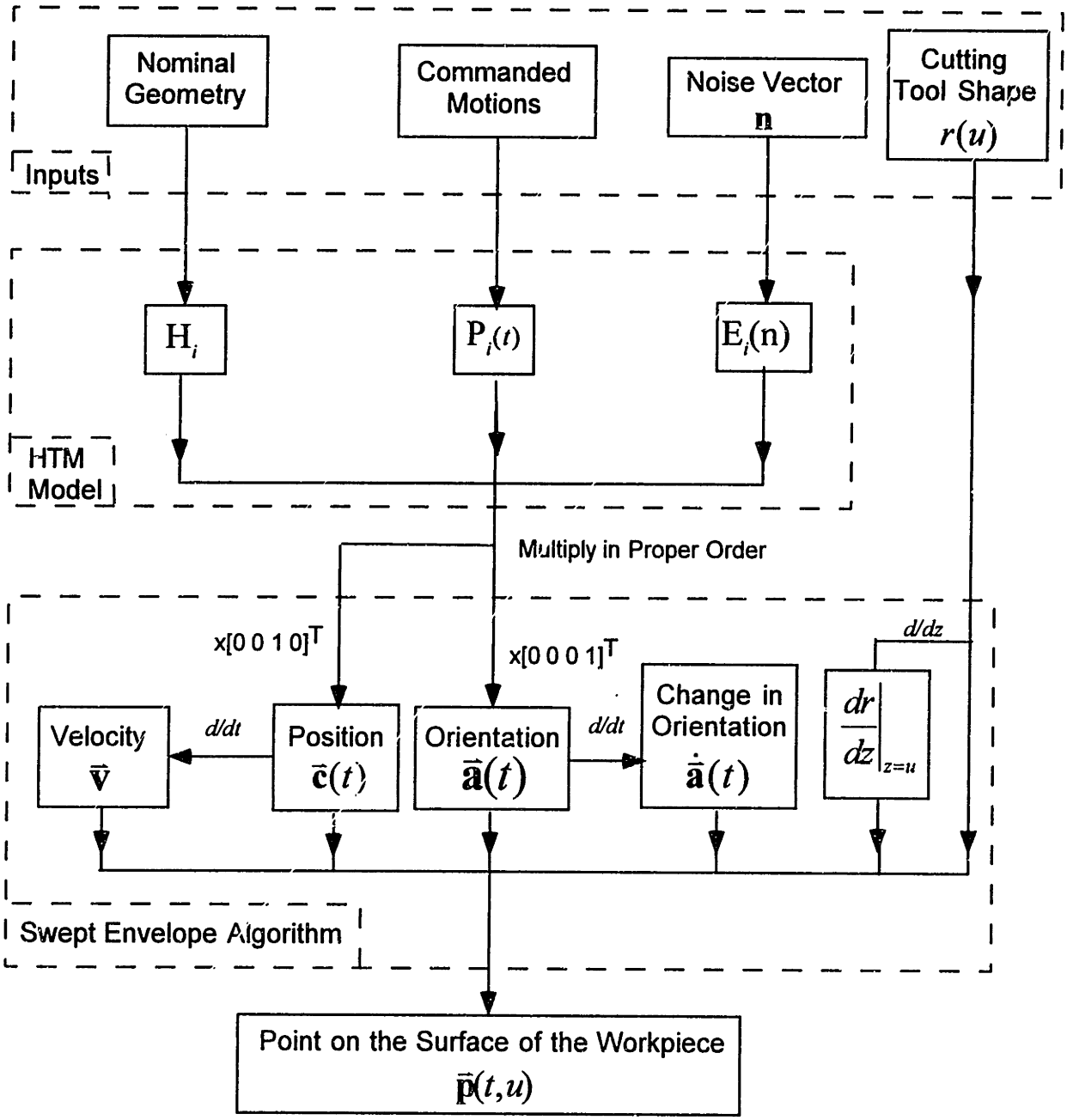


Figure 9.4.1.1 The structure of a machining simulation.

9.4.2 Virtual Parts

The simulation described in Section 9.4.1 returns one or two points $\vec{\mathbf{p}}(t,u)$ for any given value of time t and axial position on the cutting tool u. If the simulation is repeated for many discrete times t and many discrete axial positions u along the cutting tool, then one may create a set of points distributed across the surface of a simulated workpiece. Let us call this set of three

dimensional points $M(\mathbf{n})$. The M designates the fact that it is the result of a machining simulation. The result of the simulation is a function of some instance of the noise vector $\mathbf{n}$. The set of points $M(\mathbf{n})$ is an approximate description of a machined workpiece for a given set of assumed conditions described by the noise vector $\mathbf{n}$. Since this set of points represents the geometry of a workpiece based on a simulation, let us call any instance of $M(\mathbf{n})$ a *virtual part*.

If the machining simulation is performed with all the noise factors at their target levels, the result $M(\mathbf{t})$ is a set of points on the surface of the nominal workpiece.

9.4.3 Global Validity of the Results

All of the equations given above provide points on the surface of the workpiece that are locally valid. If the swept volume algorithm returns a point at time t in a machining simulation, then the point is guaranteed to be on the swept volume of the generator. However, the methods do not guarantee the global validity of the result in the context of all machining simulations. If the swept volume of the generator contains self intersections, then there will be some complications in physical interpretation of the machining simulation.

The meaning of self intersection of a swept volume of a cutting tool is illustrated graphically in Figure 9.4.3.1. At time t, the swept volume algorithm correctly determines that point $\vec{\mathbf{p}}(t,u)$ lies on the swept envelope of the cutting tool. However, at a later time $t+\Delta t$, the point $\vec{\mathbf{p}}(t,u)$ lies in the interior of the cutting tool. In other words, the NC program for this machining operation causes the swept envelope to intersect itself. There are generally two ways to deal with this problem in practice. One may simulate only the finishing pass of an NC program, or one may perform some post processing of the set $M(\mathbf{n})$ to trim superfluous points from the swept envelope. One method of detecting self intersections of swept envelopes is given by Martin and Stephenson [1990].

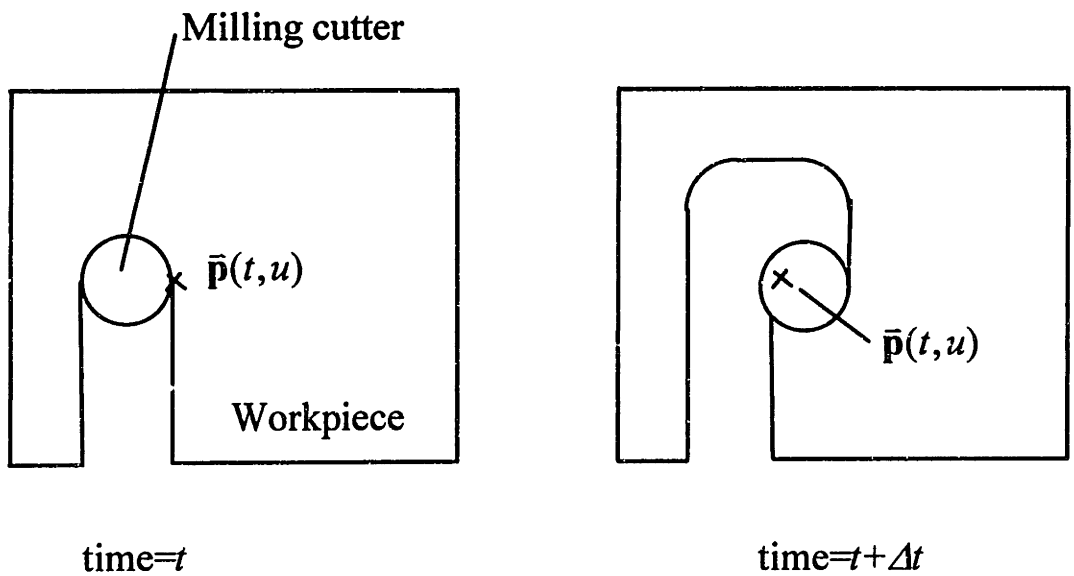


Figure 9.4.3.1 An NC program for which the swept volume result is not globally valid.

9.5 Noise signatures

There are a great many possible noise factors in machine tools. Machine components may have imperfect form, machine axes may be misaligned, drive elements are subject to thermal expansion. It has been discovered empirically that each of these noise factors has a unique "signature" which may be revealed in different machine tool operations. This section will explore this phenomenon and its relationship to virtual machining.

9.5.1 Noise signatures from Ball Bar Tests

A ballbar is a device designed for machine tool calibration and fault diagnosis. It is comprised of a length transducer with one end kinematically mounted on the machine spindle and the other end kinematically mounted on the machine bed (Fig. 9.5.1.1). In a ball bar test, the machine is programmed to describe a circle and the ballbar transducer detects deviations in the machine's tool path. Similar results can be achieved by measuring a cylindrical master artifact [Knapp, 1983, 1987].

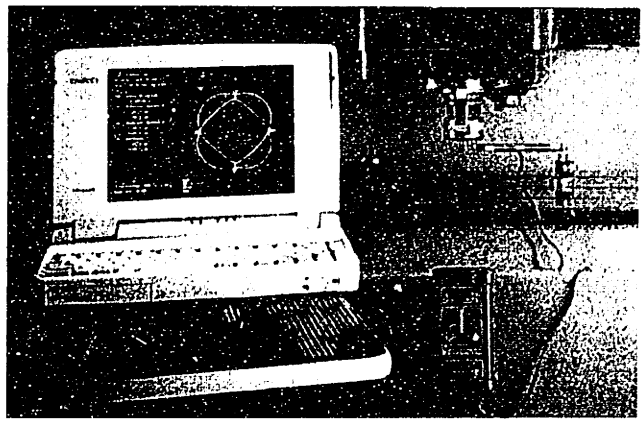


Figure 9.5.1.1 The Renishaw QC10 ball bar system.

It has been observed that different sources of machine error, in isolation, produce distinct patterns in a ball bar trace. For example, an error in squareness between the axes of an NC milling center will result in the ball bar trace plotted in Figure 9.5.1.2. A cyclic error in one of the machine's ball screws will cause the trace plotted in Figure 9.5.1.3. These noise signatures were computed using simulations of ball bar tests. Similar signatures were compiled for a host of noise factors by Kakino [1985].

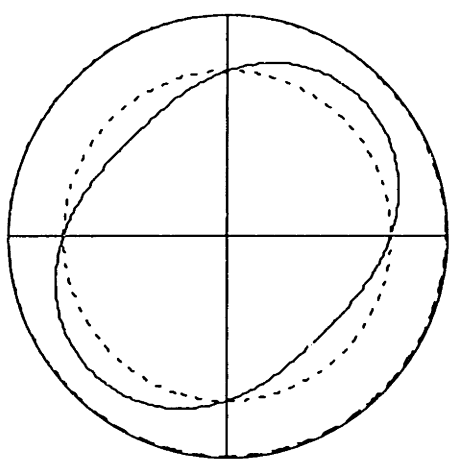


Figure 9.5.1.2 The ball bar trace resulting from a squareness error.

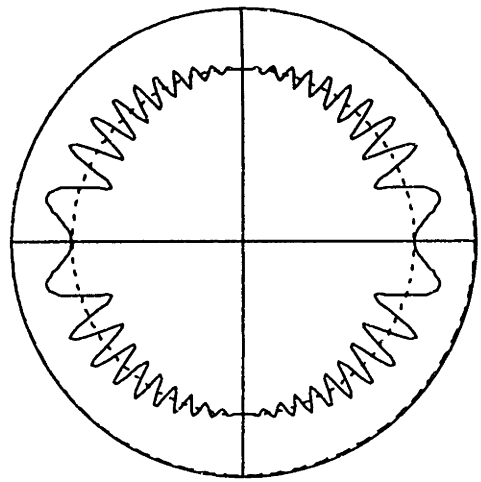


Figure 9.5.1.3 The Ballbar trace resulting from a cyclic error in a ball screw.

Noise signatures as revealed by ball bar tests have proven valuable for accuracy evaluation, calibration, and fault diagnosis. The next section introduces a generalization of the concept of noise signatures for use within the virtual machining framework.

9.5.2 Noise signatures from Machining Simulations

Using virtual machining, one may determine the noise signature associated with a noise factor. A noise signature is computed by performing a machining simulation with one noise factor off target by 3σ and all the other noise factors on target. To define this notion mathematically, I define the noise signature of the jth noise factor $\mathbf{n}_j$ as

$$S_j = M\left(\mathbf{t} + 3\sigma_j \mathbf{I}_j\right) - M(\mathbf{t}) \qquad (9.5.2.1)$$

where M is the machining simulation, $\mathbf{t}$ is the target vector, σ_j is the standard deviation of the jth noise factor, and $\mathbf{I}_j$ is the jth column vector of an n by n identity matrix.

The noise signature of a noise factor is a set of three dimensional points. Noise signatures may be displayed by amplifying the noise signature, adding it onto the nominal workpiece geometry $M(\mathbf{t})$, and plotting a surface fit to the resulting set of points.

To provide an example of the noise signature concept, consider the process of turning a cylinder and dome on an NC lathe (Fig. 9.5.2.1). A cyclic error in the x axis ballscrew would cause the distinctive noise signature plotted in Figure 9.5.2.2. A squareness error of the z axis

would tend to cause the torpedo shaped noise signature depicted in Figure 9.5.2.3. Virtual machining allows one to compute similar noise signatures for any error source of concern.

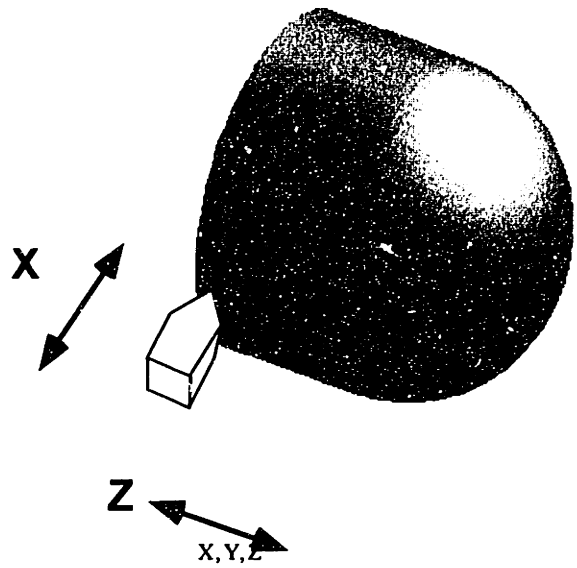


Figure 9.5.2.1 NC turning a profile, the nominal workpiece geometry.

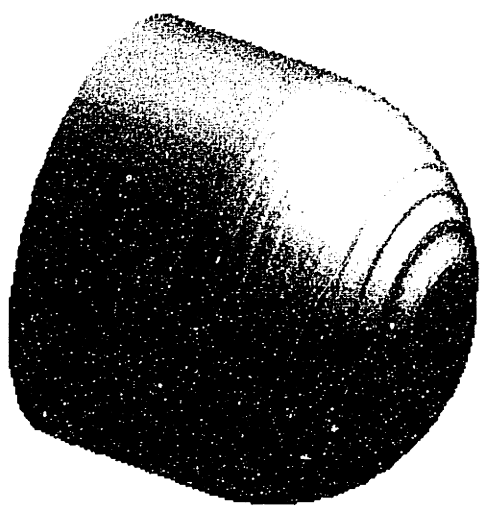


Figure 9.5.2.2 The noise signature due to cyclic error in the X axis ball screw.

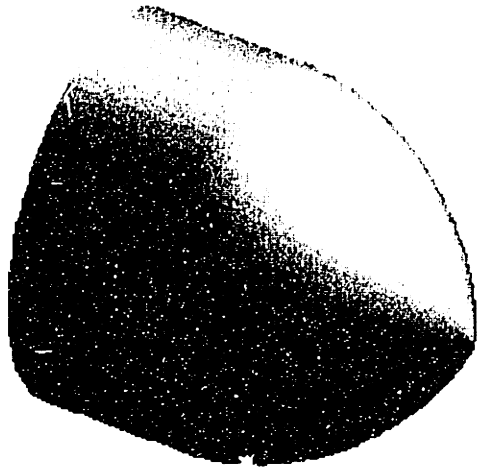


Figure 9.5.2.3 The noise signature due to a squareness error in the Z axis.

9.5.3 The Noise Signature Matrix

One of the key properties of noise signatures is that they obey superposition in the vast majority of practical scenarios. A survey of the machine tool literature reveals that the second order terms in kinematic models of machine tools are virtually always negligible compared to linear terms [Soons, 1993]. Therefore, for any sufficiently small **x**,

$$M(\mathbf{t}+\mathbf{x}) = M(\mathbf{t}) + M(\mathbf{x}) \tag{9.5.2.1}$$

The superposition principle for noise signatures leads to one of the key ideas behind virtual machining – the noise signature matrix **S**. To create the noise signature matrix, a simulation of the machining of the given part is performed for each noise factor. In other words, a matrix of points **S** is created by performing the computation of Equation 9.5.2.1 for all j from 1 to n. For each point on the nominal workpiece $M(\mathbf{t})_i$, one computes the distance from the nominal point to the perturbed workpiece surface as measured along a direction normal to the nominal surface at $M(\mathbf{t})_i$ and stores this scalar value in $\mathbf{S}_{ij}$.

Once the initial computational effort has been expended to create the noise signature matrix **S** and a nominal virtual part $M(\mathbf{t})$, virtual parts can be assembled for any noise vector with much reduced computational effort. The superposition principle of Equation 9.5.2.1 leads to

$$M(\mathbf{n})_i = M(\mathbf{t})_i + \sum_{j=1}^{n} \mathbf{S}_j \cdot \left(\frac{\mathbf{n}_j - \mathbf{t}_j}{3\sigma_j} \right) \cdot \hat{\mathbf{n}}_i \qquad (9.5.2.3)$$

where $\hat{\mathbf{n}}_i$ is the unit normal to the nominal surface at the ith point on the nominal workpiece. In other words, for any noise vector **n** within the neighborhood of the target vector **t**, the machining simulation M may be replaced by an affine transformation. This affine transformation can be far less computationally intensive than the machining simulation itself.

9.5.4 The Noise Signature Matrix and the Capability Matrix

Most of this thesis has concerned the properties and uses of the process capability matrix. For machining simulation I have now defined a different concept called an noise signature matrix. This section will consider the differences between the two and the means to compute a capability matrix from an noise signature matrix.

A column of the noise signature matrix is a representation of the geometry of a virtual part manufactured by a machine tool disturbed by a single noise factor. It is a collection of three dimensional points. A column of the process capability matrix is a listing of the normalized tolerances held on a discrete unit of product made by a manufacturing system disturbed by a single noise factor. Therefore, to create a column of the capability matrix from a column of the noise signature matrix, one simply needs to compute tolerances held as a function of a set of points sampled from the surface of a part.

The mapping from sets of points to tolerances is performed by software associated with coordinate measuring machines (CMMs), specialized inspection gages, and other metrology equipment. The mapping for some tolerances, such as cylindricity, are not standardized and are often a point of contention between CMM manufacturers and CMM users [Murthy, 1986]. However, the mapping for some of the simpler tolerances, such as roundness, are well established. I will restrict my attention to the well defined tolerances.

Table 9.5.4.1 below is taken directly from ANSI Y14.5M [1988]. It lists all of the major tolerances that can be called out within the geometric dimensioning and tolerancing standards. Let us take as a first example, the tolerance of position of a cylindrical feature. A column of the noise signature matrix will contain a set of coordinates for points about the circumference of the cylindrical feature. The location of a feature is most often taken as the least squares center of the

set of points. If the points are evenly distributed (equiangular) about the circumference of the feature and the origin is near the center of the feature, then the least squares center coordinates (x_c, y_c, z_c) are simply twice the average values of the coordinates of all the points output for the machining simulation of the feature [Murthy, 1986]

$$\begin{Bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{Bmatrix} = \frac{2}{m}\sum_{i=1}^{m} M_i(\mathbf{n}) \tag{9.5.4.1}$$

If a cylindrical feature has tolerances of position then the x and y coordinates of the least squares center meet the definition quality characteristics and they can be assigned to columns of the process capability matrix. If the x coordinate of the feature position is the ith quality characteristic and the feature is defined by points p through q in the signature matrix **S** then

$$\mathbf{C}_{ij} = \frac{1}{(U_i - L_i)/2}\frac{2}{1+q-p}\sum_{i=p}^{q}\left(S_{ij}\right)_1 \tag{9.5.4.2}$$

Similarly, if the y coordinate of the feature position is the ith quality characteristic then

$$\mathbf{C}_{ij} = \frac{1}{(U_i - L_i)/2}\frac{2}{1+q-p}\sum_{i=p}^{q}\left(S_{ij}\right)_2 \tag{9.5.4.3}$$

Inspection of Equations 9.5.4.2 and 9.5.4.3 reveals that, for positional tolerances, the mapping from the noise signature S to the process capability matrix **C** is linear. This allows one to employ the algorithms of Chapter 5 to compute rolled throughput yield.

Despite the example of positional tolerance, the mapping from part geometry to tolerance is not always linear. For example, the roundness of a cylindrical feature defined by the points p through q in a virtual part $M(\mathbf{n})$ can be computed by the formula

$$\begin{aligned} \text{roundness}\left[M(\mathbf{n})\right] = &\max\left(\left\| M(\mathbf{n}) - \frac{2}{1+q-p}\sum_{i=p}^{q} M(\mathbf{n})_i \right\|_2\right) \\ &- \min\left(\left\| M(\mathbf{n}) - \frac{2}{1+q-p}\sum_{i=p}^{q} M(\mathbf{n})_i \right\|_2\right) \end{aligned} \tag{9.5.4.4}$$

The non-linearity of the max and min functions causes the measurement of roundness to be a strongly non-linear mapping. This point requires some more detailed elucidation. The roundness error obeys one of the criteria for linearity

$$M(a\mathbf{n}) = aM(\mathbf{n}) \tag{9.5.4.5}$$

but it fails another test for linearity

$$M(\mathbf{x}+\mathbf{y}) = M(\mathbf{x}) + M(\mathbf{y}) \tag{9.5.4.6}$$

In Chapter 3, I interpreted the capability matrix as an affine mapping. This interpretation will not be valid unless *both* linearity criteria (9.5.4.5 *and* 9.4.5.6) are met. Many of the results derived in this thesis rely on the affine mapping interpretation. As I have shown above, roundness tolerances do not pass this test. To make the treatment more complete, I have categorized the commonly applied ANSI tolerances into two classes, those that exhibit linear behavior and those that do not (see Table 9.5.4.1). If any of the tolerances of a product fall in the second category, then the results developed here will tend to provide over-conservative estimates of yield. This phenomenon is explored through an industrial case study in Chapter 11.

Table 9.5.4.1 Tolerances, associated symbols, and linearity behavior classification.

Type of Tolerance	Characteristic	Symbol	ANSI Defn	Behavior
Form	Straightness		6.4.1	Non-linear
	Flatness		6.4.1	Non-linear
	Circularity (Roundness)		6.4.3	Non-linear
	Cylindricity		6.4.4	Non-linear
Profile	Profile of a Line		6.5.2b	Non-linear
	Profile of a Surface		6.5.2a	Non-linear
Orientation	Angularity		6.6.2	Linear
	Perpendicularity		6.6.4	Linear
	Parallelism - Planar		6.6.3	Linear
	Parallelism - Cylindrical		6.6.3	Non-linear
Location	Position		5.2	Linear
	Concentricity		5.11.3	Non-linear
Runout	Circular Runout		6.7.2.1	Non-linear
	Total Runout		6.7.2.2	Non-linear

9.6 Using The Results

The previous chapters provide a means to simulate the operation of a machine tool. The simulations allow one to create a noise signature matrix **S** that captures the response of a machine tool to noise factors.

9.6.1 Manual Diagnosis

Qualitative information from error shapes can allow diagnosis of their source. Kahino [1993] describes a method of diagnosing error sources in NC machines using data from ball bar tests. He first develops one error shape for each of a set of anticipated error sources. Kahino's methods for developing these error shapes are similar to that used in virtual machining. He mathematically models individual error sources to determine their effects on the error motions of rigid bodies in the machine. He then places these error motions into a kinematic model of both the machine tool and the ball bar itself to compute the error shape. The error shapes are plotted and visually compared to error traces from ball bar tests. Kahino provides a list of rules to guide the user in finding similarities in the patterns between the experimental and computed error shapes.

The machining simulation can generate a set of error traces analogous to those in Kahino's book except that these error traces can apply to any part built on any machine. Kahino's traces apply only to ball bar tests and only to NC machine tools with three orthogonal axes. Using the error shapes from our simulation, Kahino's diagnostic methods can be applied using post process gauging data instead of ball bar tests. This technique (illustrated in the case study of Chapter 11) has several advantages:

- Some errors (e.g., tool chatter) will not be evident from ball bar tests but can potentially be modeled in a machining simulation.
- In some cases, post process gauging data is collected for statistical process control. By employing the simulated error shapes, one can use already available data for the diagnosis and avoid the down-time required to perform diagnostic ball bar tests.

- Ball bar tests limit the motion of the machine to the surface of a half sphere. Employing the machining simulation approach allows more faithful reproduction of the conditions under which errors occur.

The application of Kahino's diagnostic procedures should be effective under the right conditions. The post process gauging data must have an adequate signal to noise ratio. If not, the error shape to be matched will be masked by noise. Also, the source of the errors in the parts has to be among the errors anticipated and simulated. If an error pattern arises in the parts that does not match that of any anticipated error sources, then the simulation can be used to experiment with new error sources.

A good match between simulated and measured error shapes does not guarantee that the actual source of the error has been identified. It is conceivable that some unanticipated source of error has an error shape very similar to one of the errors included in the simulation. The way to confidently diagnose the source of an error is to measure the variations in the error source and establish a statistical correlation with the magnitude of the measured error in the part.

9.6.2 Automated Diagnosis

One of the uses of a ball bar test is to diagnose faults in a machine tool. Ball bar manufacturers have written software that estimates the magnitude of various error sources in a machine tool. The software employs a database of noise signatures such as those depicted in Figures 9.5.1.2 and 9.5.1.3. In order to make a diagnosis, the software attempts to match an experimentally derived Ballbar trace with a combination of the stored noise signatures. Thus, the diagnosis software can display estimated values for each of several possible error sources.

Virtual machining extends the basic concept of noise signatures to much wider application. Simulations of machining processes can yield noise signature matrices **S** (see Section 9.4.3) not only for Ballbar tests, but for *any machining operation.* One consequence of this fact is that virtual machining can be used to create a fault diagnosis system catered to any machining process. Rather that stopping production to perform a Ballbar test, a diagnosis can be made with post process gauging data from machined parts. If the results of post process gauging are stored

in a vector **g** then the vector of noise factor values **n** that most nearly matches the gauging data is given by

$$\mathbf{n} = \left(\mathbf{S}^T\mathbf{S}\right)^{-1}\mathbf{S}^T\mathbf{g}. \tag{9.6.2.1}$$

The vector **n** can be viewed as a diagnosis of the causes of the pattern of error observed in the gauging data **g**. The elements of the vector **n** are likely values of the noise factors extant for the manufacture of the discrete unit of product characterized by the post process gauging data **g**.

There are strong cautions to be heeded in applying Equation 9.6.2.1 to diagnosis of machining problems. If two noise factors have identical effects on the workpiece, then $\mathbf{S}^T\mathbf{S}$ will be singular and the procedure will fail. If a noise factor (or a linear combination of noise factors) has an effect on the workpiece very similar to that of another noise factor, then $\mathbf{S}^T\mathbf{S}$ will be ill conditioned and the procedure will yield unreliable results.

Despite the cautions noted above, a least squares diagnosis procedure can have significant practical applications. The procedure can save valuable machine up-time. Rather than stopping production to perform a Ballbar test, one base diagnoses on gauging data that is routinely gathered as part of quality assurance programs. Not only does MVA based diagnosis save time over ball bar tests, it is potentially more complete. Post-process gauging data can be a much richer source of diagnostic information than a ball bar test. For example, gauging data from machined parts can reveal problems with part fixturing and tool wear that a ball bar test could never detect.

9.6.3 Error Budgeting

An error budget is a systematic account of the sources of error in a machine tool. It can be used to estimate the accuracy of a machine during its design. It can also help to identify the major contributors to overall machine error so that better design decisions can be made. To form an error budget, one must:

- Estimate the magnitude and statistical distribution of error sources. This difficult and important task falls outside the scope of this thesis but its importance should not be underestimated; Any results from an error budget will be only as good as the estimates of the error sources.

- Model the mechanism that converts the error source into errors in part shape (or at least relative position of the tool to the work). This task can be accomplished by the virtual simulation framework.
- Combine errors from various sources with different statistical distributions. This has traditionally been done using empirical rules of thumb. We found that these rules worked poorly for our purposes and have developed a different solution.

Slocum categorizes machine error motions as systematic, random, or hysteretic [Slocum, 1993]. Random error sources are errors that vary under "apparently equal conditions" [CIRP, 1978] either from part to part, from machine to machine, or as a function of time (depending on the context of the analysis). The term 'random' does not imply that the variation in error has no cause, only that causes have not been determined analytically. Systematic errors are those error sources that "always have the same value and sign at a given position and under given circumstances" [CIRP, 1978]. Hysteretic errors are those errors that have a nearly constant magnitude but that vary in sign with changing conditions [CIRP, 1978] (e.g., the sign of error due to backlash changes when a carriage reverses direction).

To combine these three categories of errors, Slocum outlines a procedure in which three separate sub-budgets are maintained. In the systematic sub-budget, errors are added together and the sign is preserved so cancellation may sometimes occur. The same procedure applies to the hysteresis sub-budget. In the random sub-budget, both a "best case" arithmetic summation and a "worst case" root sum of squares summation are maintained. The "best case" and "worst case" summations can be averaged to yield a compromise estimate of the random error. The sums of the three sub-budgets can be used as an estimate of machine error.

Although this procedure has demonstrated utility in estimating the *three-dimensional uncertainty* of machine tools (as defined by the CIRP [1979]), it is not effective in combining non-linear tolerances (as discussed in Section 9.5.4). Although geometry errors can be superposed linearly point-by-point, the values of form errors are not additive (not even approximately). As a result, it is difficult to develop statistical procedures or empirical rules for combining errors of form. These difficulties will be even greater for combining random errors. To solve the problem of combining errors of form, we chose to develop a Monte Carlo simulation for combining errors.

Figure 9.4.3.2 is a schematic of the operation of the Monte Carlo simulation. The tool allows the user to define the probability distribution function (pdf) of various sources of error. For a designated number of trial parts, the program randomly samples from the user defined pdf's to determine the magnitude of the error in each individual part. The program uses these source magnitudes to linearly superpose the source error shapes drawn from a data base. The result is an error shape for that particular trial part. This error shape is processed to determine errors in size, form and profile. The values of the virtual part errors are stored for each trial and can later be used to characterize the statistical distribution of the errors in several ways. For example, histograms of the error distributions can be displayed, mean and standard deviation can be calculated, and the percentage of parts out of tolerance limits can be estimated.

The Monte Carlo error combination program should be valuable to machine designers in rationally allocating build tolerances for machine tools and other manufacturing equipment such as robots and transfer line equipment. It may also be valuable to engineering managers and engineers in specifying the allowable variance in any variables known to affect the accuracy of a manufacturing process (e.g., ambient temperature, variance of geometry that mates with a fixture, coolant flow rates, etc.). The algorithm is demonstrated in the context of an industrial case study in Section 11.8.

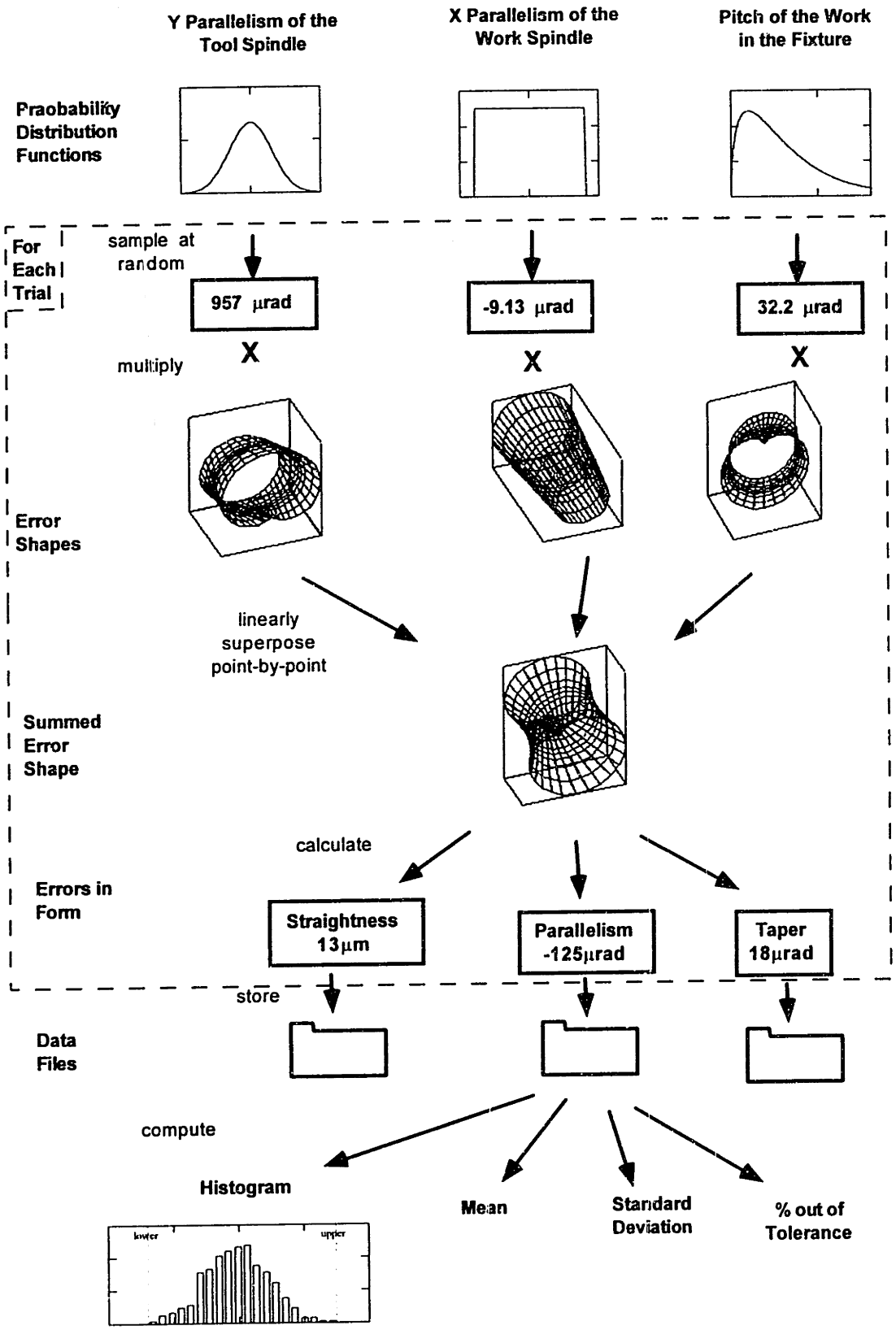


Figure 9.6.3.2 The Monte Carlo algorithm for machining.

9.7 Chapter Conclusions

This chapter has introduced techniques for simulating machining processes. The construction of a simulation begins with a kinematic model of a machine tool's carriages. This provides a motion function as input to a swept envelope algorithm. The swept envelope algorithm creates sets of points on the surface of the workpiece. These sets of points may be stored for use in fault diagnosis and error budgeting. The next two chapters discuss applications of the virtual machining techniques and the capability matrix approach to producibility analysis.

10. Dual Head Valve Grinding

This chapter presents a case study of producibilty analysis for a dual head valve grinding machine. The chapter will motivate the study by placing it in the context of the automotive industry. It will then discuss the details of the modeling effort. Finally, it will discuss the use of the model for design of the production equipment. This study will reveal some limitations to the capability matrix approach to producibility analysis arising from fundamental non-linearities in the machine's response to noise factors..

10.1 Introduction to Dual Head Valve Grinding

Intake and exhaust valves for internal combustion engines often have fairly complicated profiles. They also require tight tolerances on profile, especially on the valve seat (the face that mates with the cylinder head). Figure 10.1.1 is an example of a typical valve profile including tolerances.

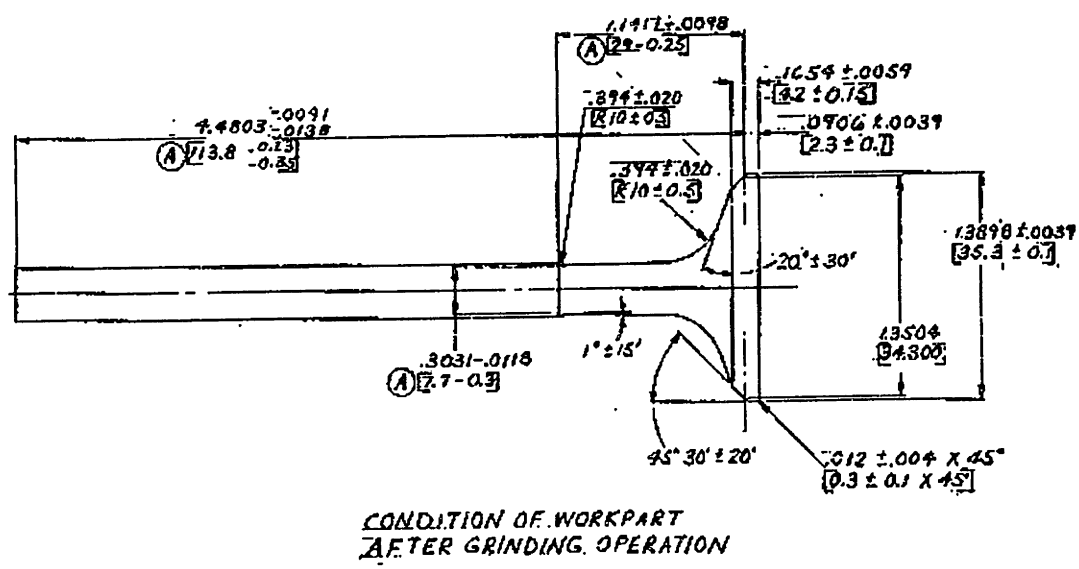


Figure 10.1.1 Dimensions and tolerances on an intake valve.

To achieve complex profiles, tight tolerances, and high production rates, form grinding is usually performed. On some "dual head" valve grinding machines, production rates are improved by grinding two valves simultaneously on the same machine (Fig. 10.1.2). The valves

are offset vertically from the horizontal plane that contains both the grinding wheel axis of rotation and the wheel in-feed axis. The valve axes of rotation are horizontal and the grinding wheel plunges in at an angle so that the grinding wheel axis of rotation is skew to both the valve axes of rotation. The contact between the grinding wheel and valve therefore occurs along a three dimensional space curve. This complicates many analyses of the machine. For example, there is no straightforward mapping between the desired valve profile and the proper grinding wheel profile. Even if the proper profile of the grinding wheel is known, there is no simple formula for determining the effect of machine errors on the profile of the valve. Machine designers need to estimate the sensitivity of machine errors (such as wear of the grinding wheel, fixturing errors, and spindle error motions) on the dimensions of the part. These estimations often form the basis of a machine error budget. The methods of computing swept volumes of surfaces of revolution are useful for solving all of the above problems.

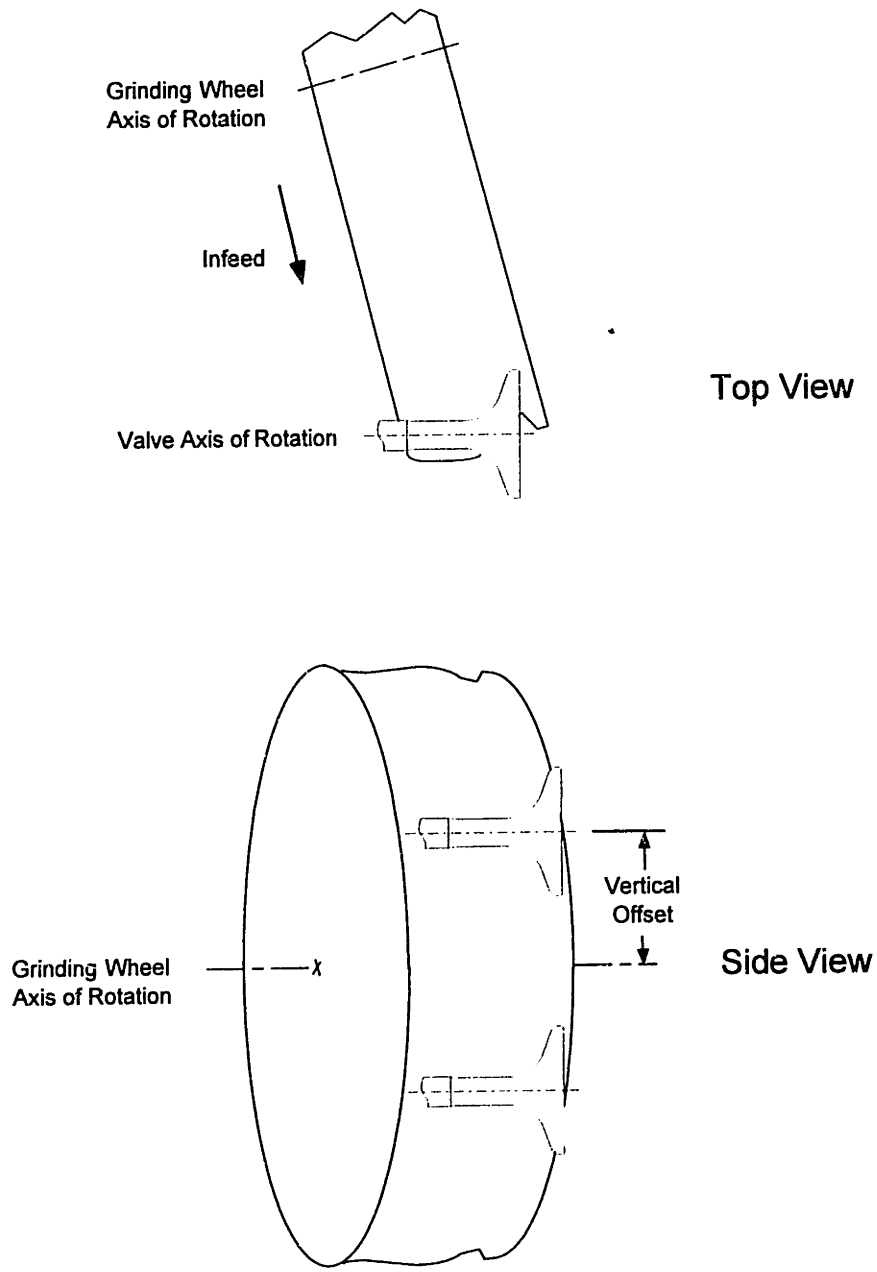


Figure 10.1.2 Schematic diagram of dual head valve grinding. The size of the grinding wheel compared to the valve is reduced for clarity.

10.2 Representing the Desired Valve Profile

I simplified the valve profile of Figure 10.1.1 to that shown in Figure 10.2.1 to make verification of our results easier for the reader. I then represented the simplified profile of the valve as a piece-wise linear or circular function of axial position. A parametric representation of the profile

would become necessary if there existed any undercuts in the profile. In this case, the profile may be represented as a function of axial position as in Equation 10.2.1.

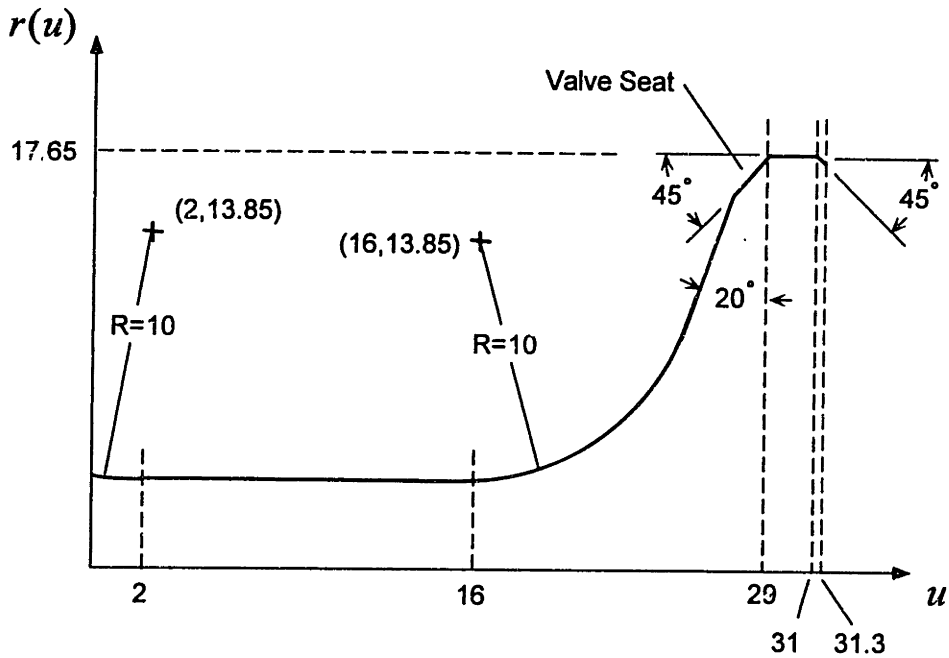


Figure 10.2.1 A simplified valve profile.

$$r(u) = \begin{cases} 13.85 - \sqrt{10-(u-2)^2} & \text{if } 0 \le u \le 2 \\ 3.85 & \text{if } 2 < u \le 16 \\ 13.85 - \sqrt{10-(u-16)^2} & \text{if } 16 < u \le 16 + 10\cos(20^\circ) \\ 13.85 - 10\sin(20^\circ) + \dfrac{u-(16+10\cos(20^\circ))}{\tan(20^\circ)} & \text{if } 16 + 10\cos(20^\circ) < u \le 27.46684001665 \\ 17.65 + (z-29) & \text{if } 27.46684001665 < u \le 29 \\ 17.65 & \text{if } 29 < u \le 31 \\ 17.65 + \dfrac{17.35-17.65}{31.3-31}(u\text{-}31) & \text{if } 31 < u \le 31.3 \end{cases} \tag{10.2.1}$$

10.3 Computing the Grinding Wheel Profile

The proper grinding wheel profile was generated by a simulation of a dressing process. Imagine that there exists a dressing wheel that is shaped, positioned, and oriented just like one of the valves. This imaginary dresser was treated as the cutting tool while the grinding wheel was treated as the workpiece (Fig. 10.3.1). The workpiece (grinding wheel) LCS is rotated 15 degrees with respect to the tool (dresser) LCS, offset by an unknown amount in its local x

direction, and offset in the negative y direction by 90mm. The work (grinding wheel) also rotates about its own z axis by an angle $\theta(t)$ that varies over time. The tool (dresser) LCS is taken as the base coordinate system so that ${}_0\mathbf{T}_m$ is simply the identity matrix. The HTM describing the position of the workpiece (grinding wheel) with respect to the base coordinate system is:

$$ {}_0\mathbf{W}_n = \begin{bmatrix} \cos(15°) & 0 & \sin(15°) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(15°) & 0 & \cos(15°) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & x_{offset} \\ 0 & 1 & 0 & 90mm \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\theta(t)) & -\sin(\theta(t)) & 0 & 0 \\ \sin(\theta(t)) & \cos(\theta(t)) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10.3.1) $$

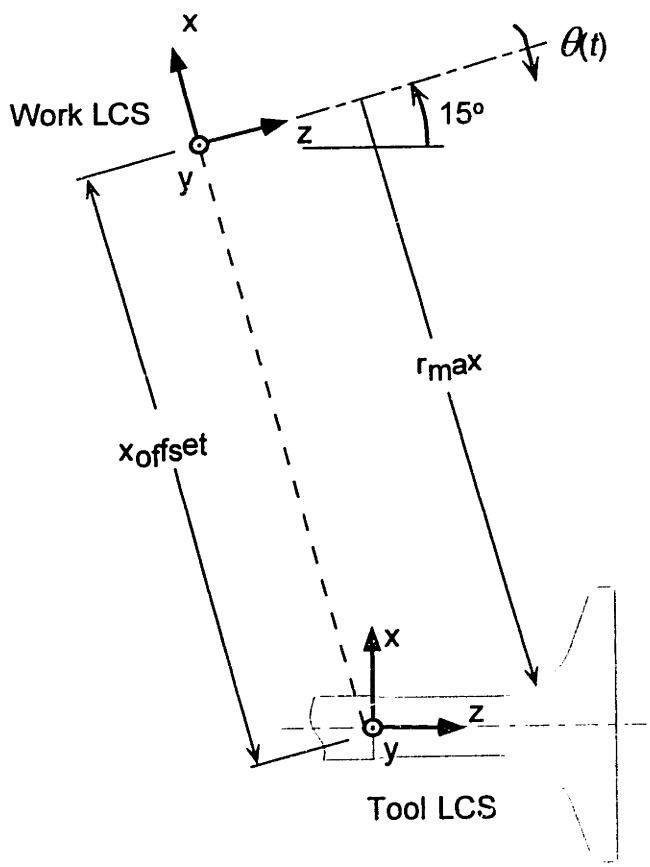


Figure 10.3.1 Arrangement of local coordinate systems for grinding wheel profile computation.

Applying the methods of this thesis, one can solve for the grinding wheel profile given any x_{offset}. Equation 9.4.2.14 yields values of $\vec{\mathbf{p}}$. The axial position z is simply the z component of $\vec{\mathbf{p}}$. The radius is the distance of $\vec{\mathbf{p}}$ from the z axis. Since the calculations can be implemented in numerical computation packages, it is a simple matter to use the secant method to solve for the x_{offset} that results in a grinding wheel with a maximum diameter of 381mm. The required x_{offset} is 369.578780mm. Some of the computed radial values are provided in Table 10.3.1.

Table 10.3.1 A sampling of computed grinding wheel radius values.

u(mm)	rw(u)(mm)
1.023801	376.459311
2.901922	377.156728
16.425335	380.678655
18.919197	381.000000
27.032058	376.782603
32.397625	370.610618
34.395383	371.100392
34.660725	371.511278

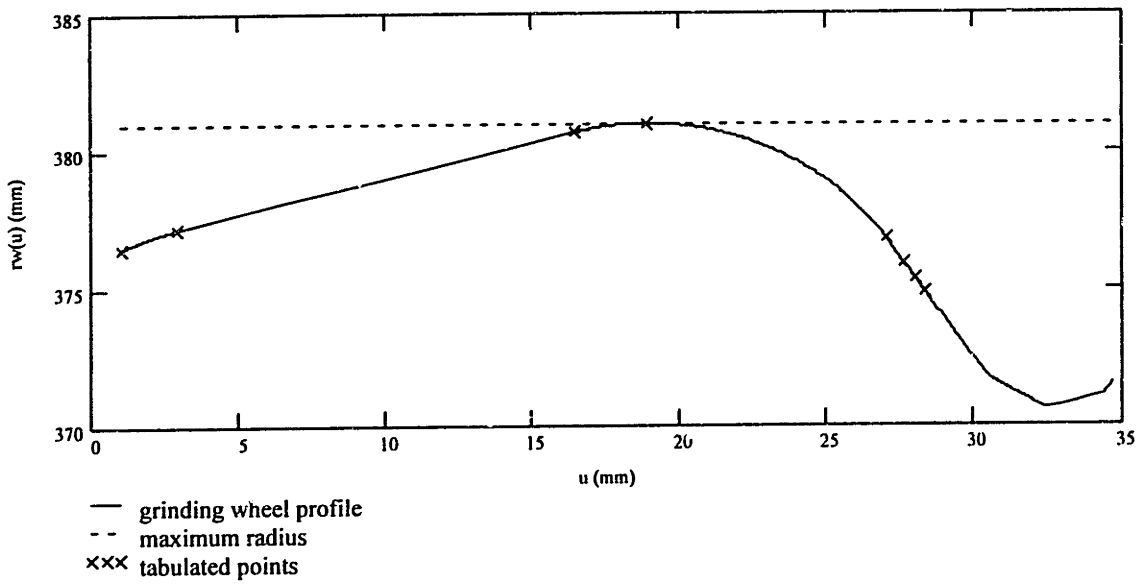


Figure 10.3.2 Radius of the grinding wheel versus axial location in the grinding wheel LCS.

The results in Table 10.3.1 and Figure 10.3.2 have been verified with FastSURF, a surface modeling CAD tool. I constructed a NURBS surface that represents the desired valve profile. Then, for each of the tabulated pairs of z and r values, I constructed a circle representing a cross section of the grinding wheel with the proper location and orientation. In all cases, the circle intersected the surface while a circle with a radius one nanometer smaller did not intersect the surface. Although it is straightforward to *verify* the results of Table 1 using commercially available CAD programs, I know of no way to *compute* the same results through a geometric construction with a commercially available CAD program (without manual iteration).

10.4 Non-linearity and Error Budgeting

The modeling techniques introduced in Chapter 9 may be extended to error analysis of the dual head valve grinding machine. As an example, consider the effect of grinding wheel wear. The grinding wheel must be dressed periodically to maintain its profile and expose new cutting surfaces. Therefore, the radius of the wheel is constantly decreasing. The proper profile of the grinding wheel changes as the radius decreases. Using the methods presented above, the proper profile could be computed based on the current radius and generated with a CNC dresser. However, this may not be economically viable. In practice, it is more likely that a formed dressing wheel will be used and that some profile error will be tolerated as the grinding wheel

wears. This profile error must be budgeted along with other errors that are likely to exist in the machine and fixtures to determine if the sum of the errors will produce acceptable parts.

To determine the effect of grinding wheel wear on the dimensions of the valve, I assumed that a dressing wheel exists with the profile of Figure 10.3.1. The profile of a worn grinding wheel after dressing is therefore the profile of Figure 10.2.1 minus some constant wear value. Taking this worn profile as the profile of the dressing wheel, I constructed a kinematic model of the machine. The resulting HTM is similar to the one given by equation 10.3.1 above except that the grinding wheel is the cutting tool and the valve is the workpiece. Also, in this model, the valve must rotate about its own z axis. The HTM model is therefore:

$$ {}_0\mathbf{W}_n = \begin{bmatrix} \cos(\theta(t)) & -\sin(\theta(t)) & 0 & 0 \\ \sin(\theta(t)) & \cos(\theta(t)) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10.4.1} $$

$$ {}_0\mathbf{T}_m = \begin{bmatrix} \cos(15^\circ) & 0 & \sin(15^\circ) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(15^\circ) & 0 & \cos(15^\circ) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & x_{offset} \\ 0 & 1 & 0 & 90mm \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10.4.2} $$

The variable x_{offset} represents the wheel in-feed. It should be adjusted to remove the size error resulting from dressing the wheel. We assumed that in-process gauging is employed at the z=29mm location on the valve and that in-feed is adjusted to reduce size error at this point to zero. This is simulated by using a search to find the x_{offset} value that forces the calculated value of the valve radius including error to equal the radius of the original valve profile at z=29mm.

One of the most critical features on the valve is the valve seat (see Figure 10.2.1). This part of the profile lies between z=27.467mm and z=29mm on the original valve profile. Figure 10.4.1 shows that, as the grinding wheel wears, the angular error on the valve increases. The non-linear shape of the curve is significant in this context. It is frequently assumed in machine error analysis that an error sensitivity or gain is sufficient for error budgeting. In this case a gain value

determined by evaluating the effect of a small error is the slope of the linear approximation in Figure 12. Based upon the linear model, the designer would estimate that a radial attrition of 128*mm* is acceptable (all other error being comparatively small). However, due to the concavity of the function, the grinding wheel can certainly not be used after more than 89*mm* of radial attrition. Of course, the limit should be set to somewhat less than 89*mm* to allow for other errors that may simultaneously exist in the machine.

The extremely wide range of the noise factor of grinding wheel attrition gives rise to the non-linearity evident in Figure 10.4.1. This level of non-linearity adds unacceptably large modeling error were one to apply the capability matrix methods of Chapters 3-8. For example, if one models the radial attrition as a uniformly distributed random variable with a range of 0 to 100mm and apply the capability matrix method, one will estimate a rolled throughput yield of 100%. However, due to non-linearity of the system, the rolled throughput yield will actually be 89%. Clearly, the assumption of non-linearity of the system response to noise can lead to substantial error in this case. Therefore, the dual head case study helps to define the limits of the capability matrix method and place it in a more realistic context.

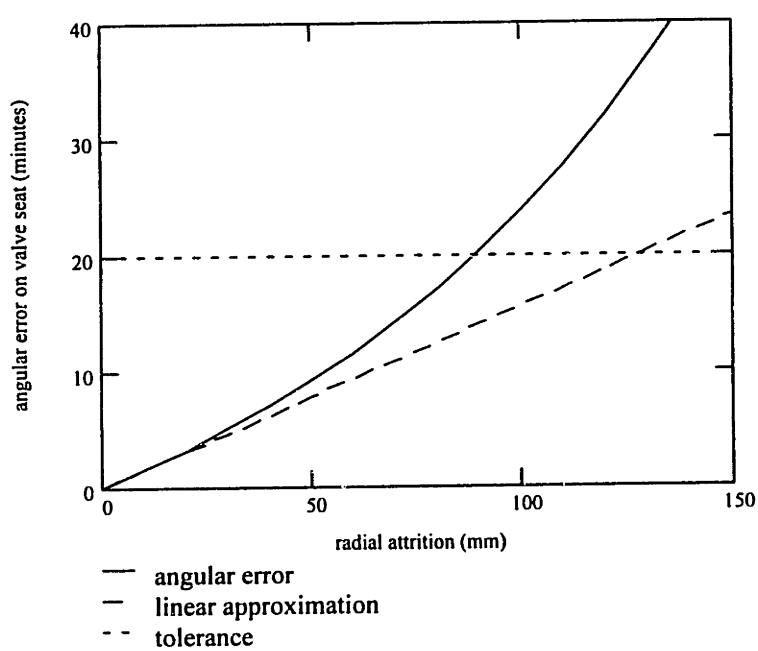


Figure 10.4.1 Determining the limit on grinding wheel wear.

10.5 Chapter Conclusions

This chapter has examined an industrial case study of producibilty analysis and error budgeting. A dual head valve grinding machine which finishes poppet valves for IC engines was analyzed. A model of the machine was developed that represented the kinematics of the machine and the geometric interaction of the formed grinding wheel with the workpiece. The study showed that there is a significantly non-linear effect due to redressing of the grinding wheel. Thus, the case study reveals an instance in which a capability matrix analysis is not sufficient for good design decision making.

11. CNC Crankpin Grinding

This chapter presents a case study of producibilty analysis for a computer numerically controlled (CNC) crankpin grinding machine. I will motivate the study by providing some background on the purpose, function, and structure of the machine. I will then discuss the construction of a "virtual machining" model of the crankpin grinder. Finally, I will discuss the use of the model for tolerance design of the production equipment. This study will reveal some limitations to the capability matrix approach to producibility analysis related to form tolerance estimation.

11.1 Quality Characteristics for Crankpins

The purpose of an engine crankshaft is to convert the vertical motion of the pistons into rotary motion. As an example, consider the V6 engine crankshaft that will be used in this case study (see Figure 11.1.1). Main journals support the shaft as it rotates with respect to the engine block. The main journal bearings are assigned Arabic numerals from left to right. Crankpins arranged eccentrically about the shaft centerline serve as journal bearings for the connecting rods that transfer forces between the pistons and the crankshaft. The crankpins are assigned Roman numerals from left to right.

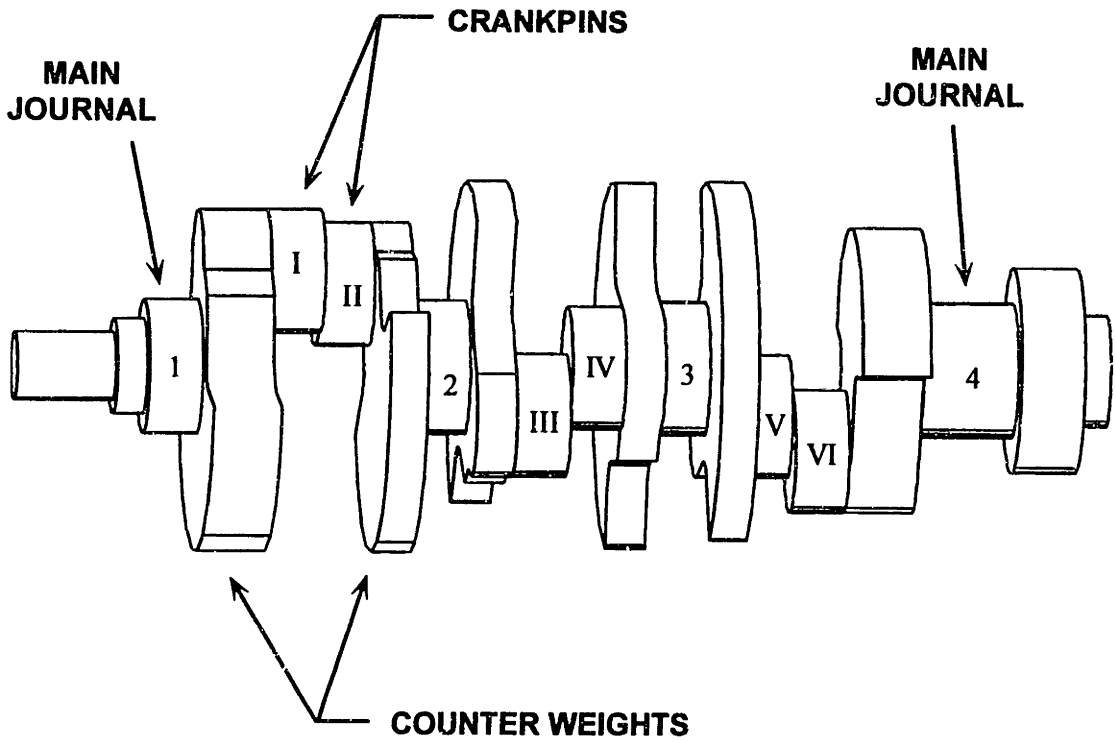


Figure 11.1.1 A V6 engine crankshaft.

The accuracy of the crankpins affects engine reliability, balance, fuel economy, and emissions. Engine performance demands are escalating. Consumers demand greater reliability, and lower NVH (noise, vibration, and harshness). Corporate Average Fuel Economy Standards obligate engine manufactures to seek better fuel economy. Federal and State environmental regulations demand that engine emissions be reduced. One of the keys to meeting these demands is tighter tolerances on engine parts in general and on crankshafts in particular.

Several tolerances are frequently applied to the crankpins. Some of the most common tolerances are listed and described below:

- Throw– A tolerance on throw specifies the maximum error in perpendicular distance of the center of the crankpin to the center of the main journals (see Figure 11.1.2).
- Index – A tolerance on index specifies the maximum error in angular measure of the crankpin from the number one crankpin (see Figure 11.1.2). There is therefore no tolerance placed on the index of the number one crankpin.

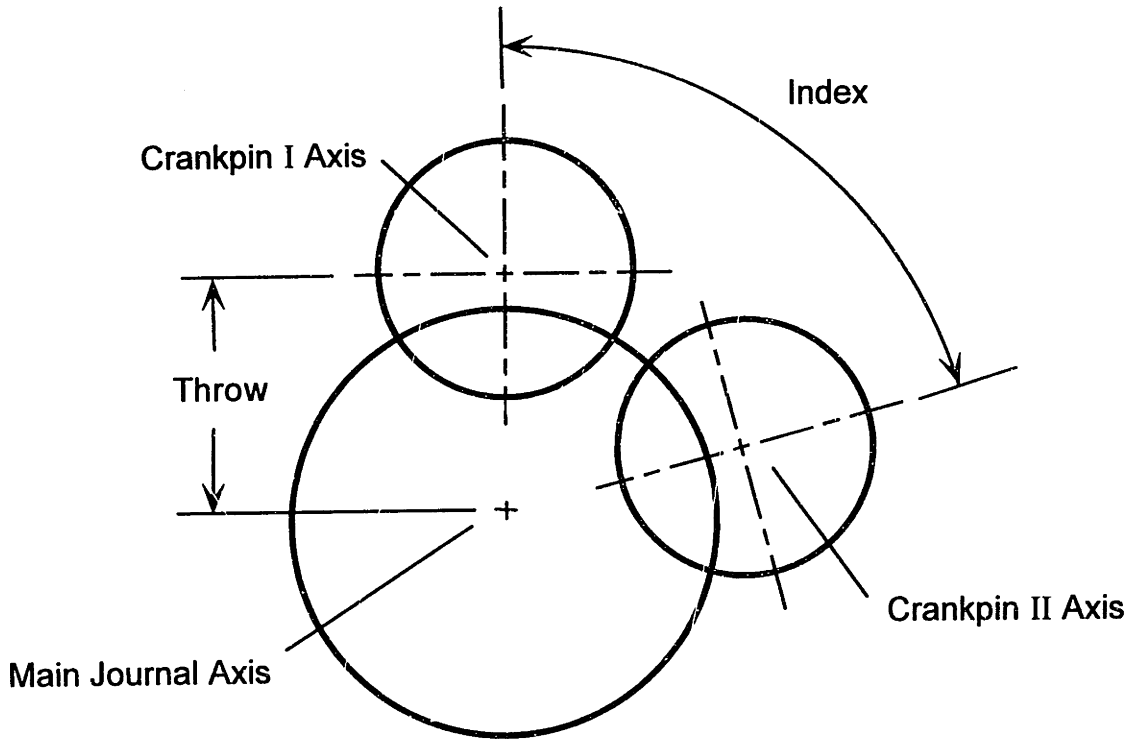


Figure 11.1.2 Errors of location for crankpins (throw error and index error).

- Size – The limits of size of a feature prescribe the extent to which variations in its geometric form are allowed [ANSI Y14.5M 2.7.1]. For a circular feature, a size tolerance specifies a circular tolerance zone within which each element of the surface must lie.
- Diameter – Although ANSI defines size as a circular tolerance zone, industry standard measuring equipment instead reports the diameter of the circle that best fits the measurement data in the least squares sense. This definition of diametral error is, in effect, a mean value; It is twice the average value of radial error measured at a multiplicity of points about the crankpin.
- Roundness – A roundness tolerance specifies a tolerance zone bounded by two concentric circles within which each element of the surface must lie [ANSI Y14.5M 6.4.3]. The tolerance on roundness must therefore be tighter than the tolerance on size.
- Conical Taper – The difference in the diameters of two sections (perpendicular to the axis) of a cone to the distance between the sections [ANSI Y14.5M 2.13].

- Parallelism – A parallelism tolerance defines a cylindrical tolerance zone whose axis is parallel to a datum axis [ANSI Y14.5M 6.6.3] (see Figure 11.1.3). Alternately, a parallelism tolerance may be defined by two parallel bounding planes.
- X Parallelism – The minimum spacing between two bounding planes normal to a vector that passes from the main journal axis through the number I crankpin axis.
- Y Parallelism – The minimum spacing between two bounding planes normal to the X parallelism tolerance planes and parallel. To the main journal axis.

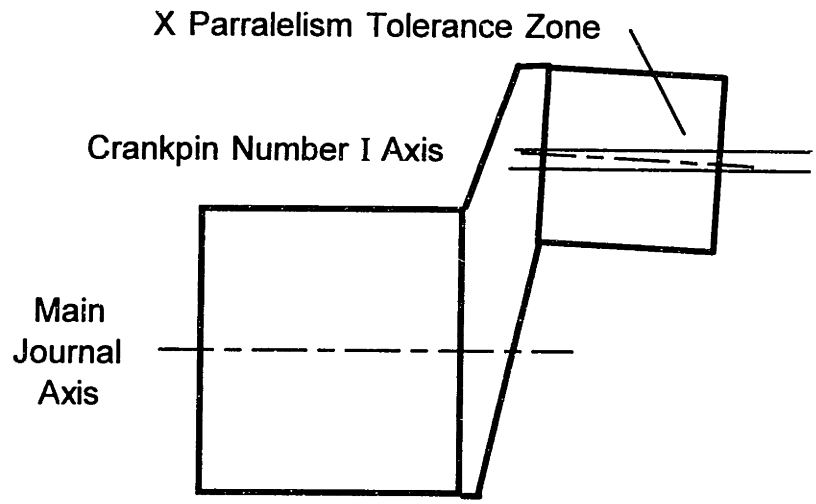


Figure 11.1.3 X Parallelism tolerance specified on a crankpin.

11.2 The CNC Crankpin Grinding Process

Consumers demand not only better quality and performance but greater variety as well. Over the past ten years, the number of auto models on the market has steadily risen while the number of cars sold per model has dropped. The need for manufacturers to make more types of cars in the same number of factories has driven a need for greater agility in manufacturing equipment and operations.

The simultaneous demands for greater agility and high precision has motivated the development of CNC grinding machines. On this type of machine, the crankshaft rotates about the main journal centerline and the grinding wheel reciprocates under numerical control to

maintain the proper position with respect to the crankpin (see Figure 11.2.1). This design makes the machine agile; A switch from grinding one type of crankshaft to another can potentially be made entirely through software; No tooling changes or adjustments are required in principle.

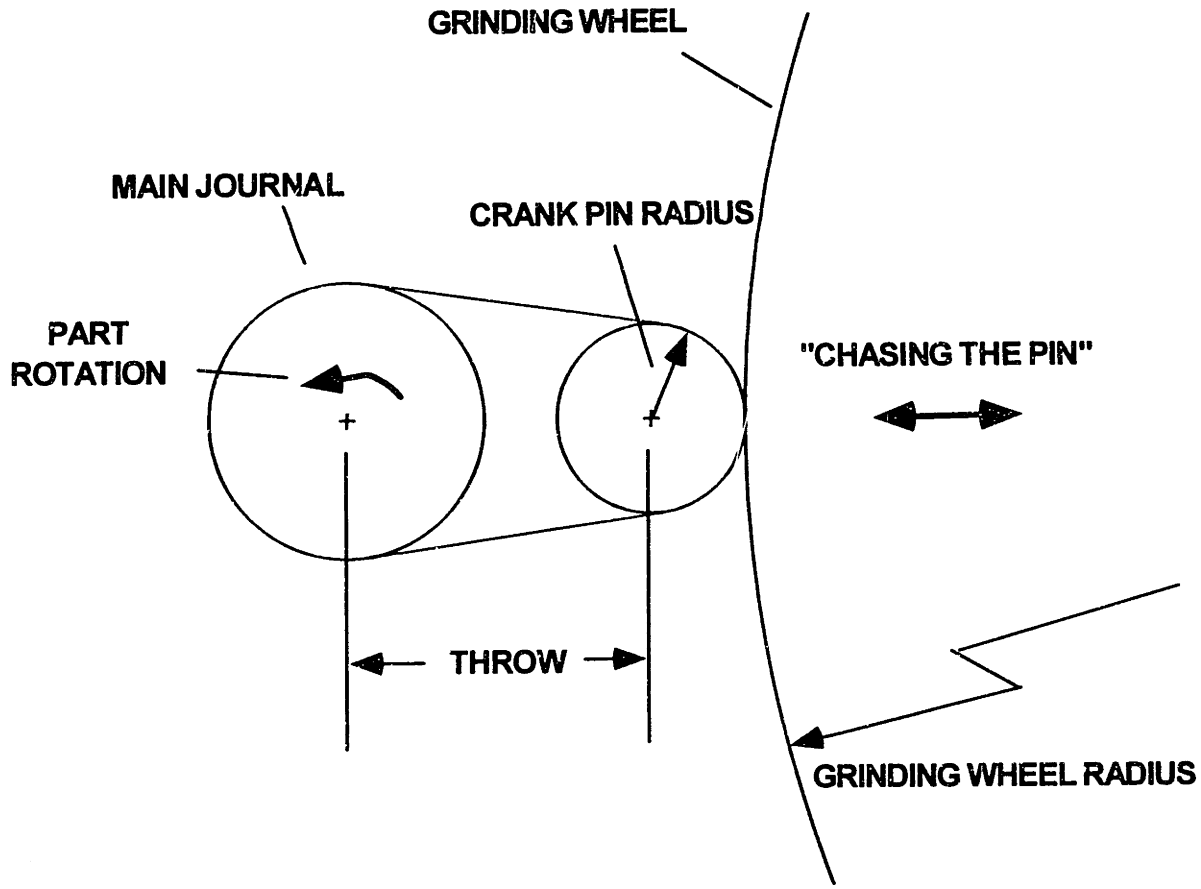


Figure 11.2.1 CNC "chasing the pin" crankpin grinding.

Error mapping techniques allow CNC grinding machines to achieve micron level accuracy; Most of the repeatable errors in the size, roundness, index, and throw of the crankpins can be compensated by NC control. However, many errors in taper, parallelism, and center deviation cannot be compensated on most CNC grinding machines. This necessitates rigorous error budgeting.

The advantages of CNC grinding come at the expense of mathematical complexity. In the CNC grinding machine most noise factors affect more than one of the tolerances called out on the crankpins. For example, wear of the grinding wheel can cause errors in size, throw, and

roundness. The coupling of the noise factors and tolerances makes error diagnosis and error budgeting more challenging.

11.3 The CNC Crankpin Grinding Machine

The machine to be modeled is depicted graphically in Figure 11.3.1. Note the coordinate definitions in the lower left hand corner of the figure. In accordance with the conventions in machine design, the z axis is the axis of the cutting tool spindle. The positive z direction is to the right in Figure 11.3.1. The positive Y direction is up. The positive x direction runs from the z axis through the grinding wheel axis. A negative x displacement of the grinding wheel will remove material from the work.

The cast iron base of the machine supports two numerically controlled carriages. The x axis carriage supports the tool spindle that drives the grinding wheel. The x axis carriage reciprocates, driven by a ball screw, as the work spindles rotates. The x axis infeed $x(a)$ is slaved to the work spindle angle a

$$x(a) = r_c + r_w + throw(1 - \cos(a)) + \sqrt{(r_c + r_w)^2 - throw^2 \cdot \sin(a)^2} \tag{11.3.1}$$

where r_c is the radius of the crankpin, r_w is the radius of the grinding wheel, and *throw* is the nominal throw of the crankshaft.

The z axis carriage supports two work spindles and is used to bring each crankpin, in turn, into alignment with the grinding wheel. Both work spindles are driven by electric motors. The two motors are synchronized to turn the #1 and #4 main bearings at the same rate.

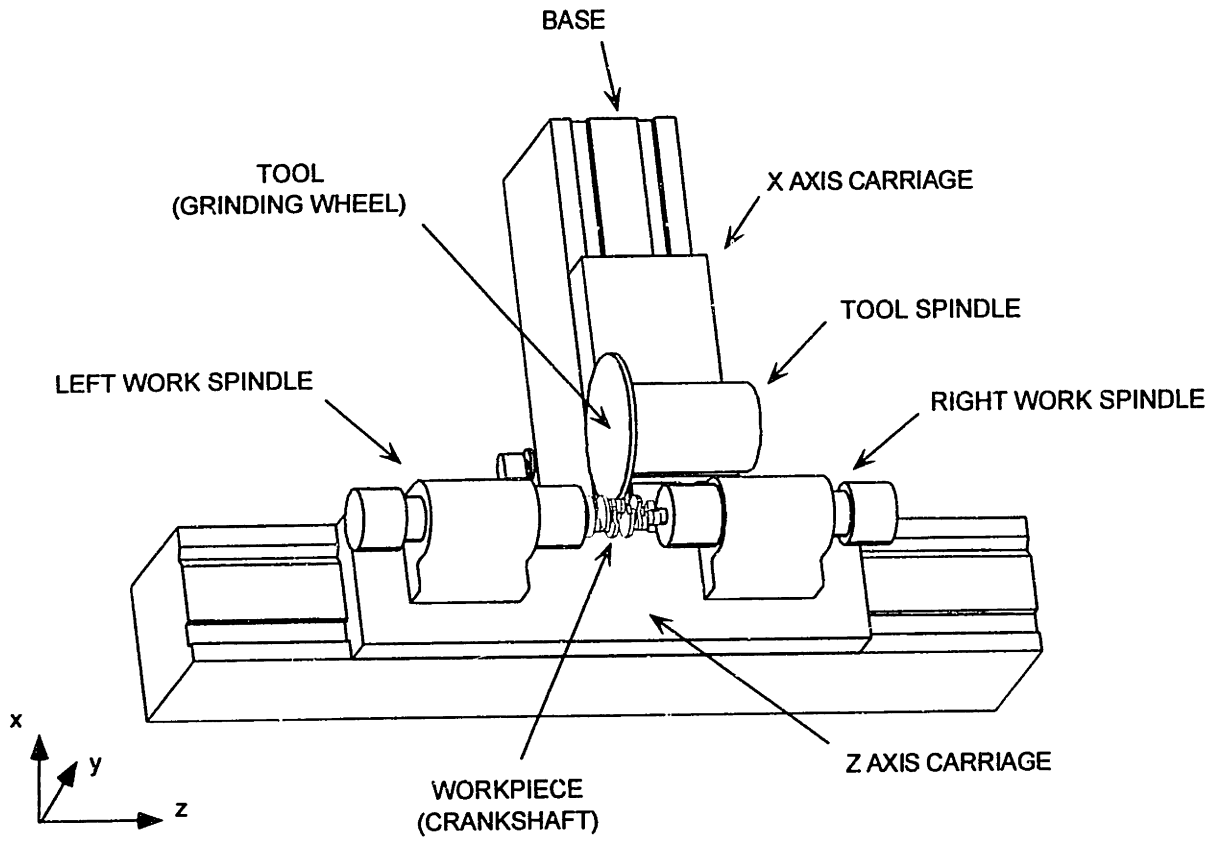


Figure 11.3.1 The CNC crankpin grinding machine and associated terminology.

11.4 Noise Factors for CNC Crankpin Grinding

A list of possible noise factors for the CNC grinding machine was formed through discussions with machine designers, machine operators, and R&D engineers. Each possible noise factor was incorporated in the HTM model to be discussed in Section 11.5 and listed in detail in Appendix E. Some of the most interesting noise factors are listed and briefly described below. Each of these noise factors has an associated noise signature plotted in Appendix F.

- Grinding wheel attrition – The grinding wheel is consumed during use through wear and redressing. If the exact change in grinding wheel size were known, the controller would be able to compensate for this effect. However, the change in grinding wheel size cannot be measured exactly resulting in errors in the roundness of the crankpins (see Figure 11.4.1). I

assume that the error in size due to attrition is corrected for with x axis infeed and that the controller is not properly updated with the new grinding wheel radius.

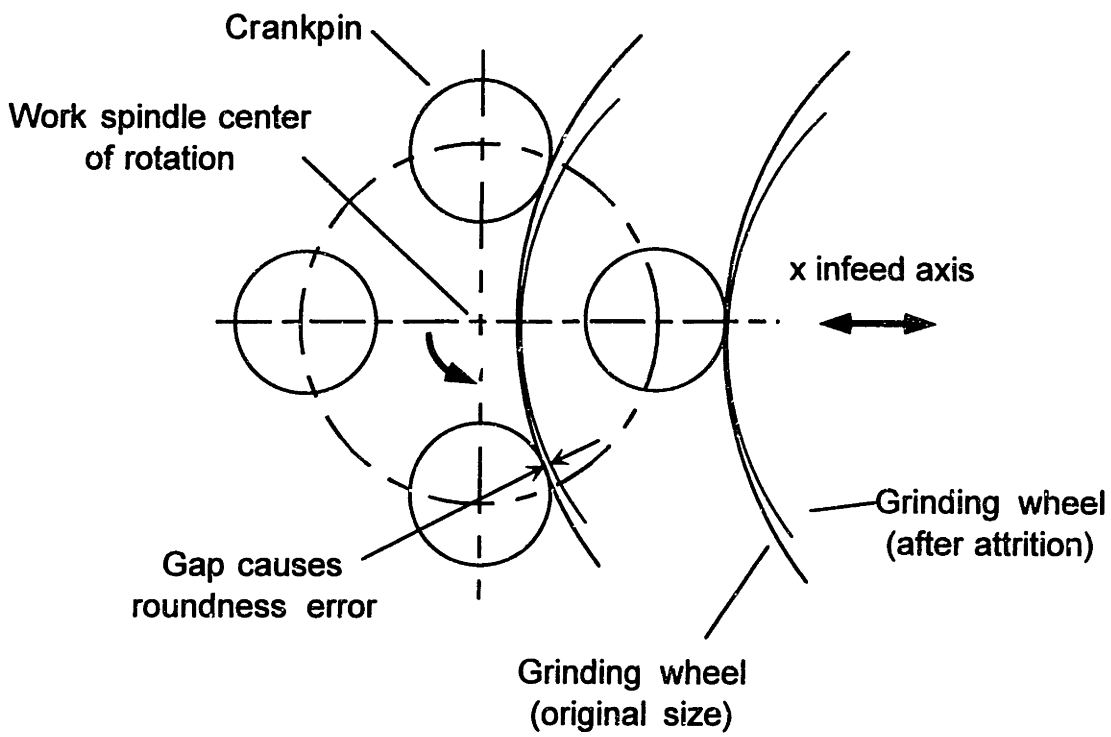


Figure 11.4.2 The effect of grinding wheel attrition.

- Main bearing size error – During the crankpin grinding operation, the workpiece is supported by its #1 and #4 main bearings. To permit rapid fixturing, the main bearings are fixtured in in 'V' blocks (see Fig. 11.4.2). Therefore, errors in main bearing size, result in a fixed offset of the main bearing centerline and the work spindle axis. As the work spindle axis rotates, the crankshaft will wobble slightly.

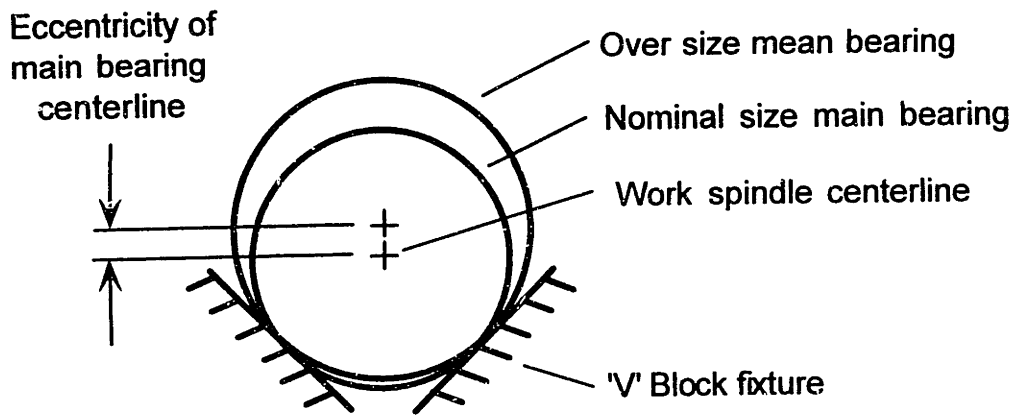


Figure 11.4.2 The effect of main bearing size error.

- Main bearing size mismatch – Due the effect of main bearing size error (see bullet above), if the #1 and #4 main bearings are not the same size, then the crankshaft main bearing axis will not be aligned with the work spindle axis. As the work spindle axis rotates, the crankshaft will precess slightly.
- Cumulative lead error in the x axis – A lead screw is designed to advance a carriage by a nominal distance per turn. In practice, lead screws will (on average) advance slightly more or less than this nominal distance (see Figure 11.4.3). This cumulative lead error is limited by manufacturer's specifications. Cumulative lead error in the x axis carriage of the CNC crankpin grinder will affect the roundness and position of the crankpins.
- Once-per-evolution lead error – A ball screw is designed to advance a carriage by a fixed amount per degree of turn. In practice, ball screws exhibit some variation from this ideal. This variation is often nearly periodic, repeating with each full turn of the lead screw (see Figure 11.4.3). This once-per-evolution lead error is limited by manufacturer's specifications. Once-per-evolution lead error in the x axis carriage of the CNC crankpin grinder will affect the roundness of the crankpins.

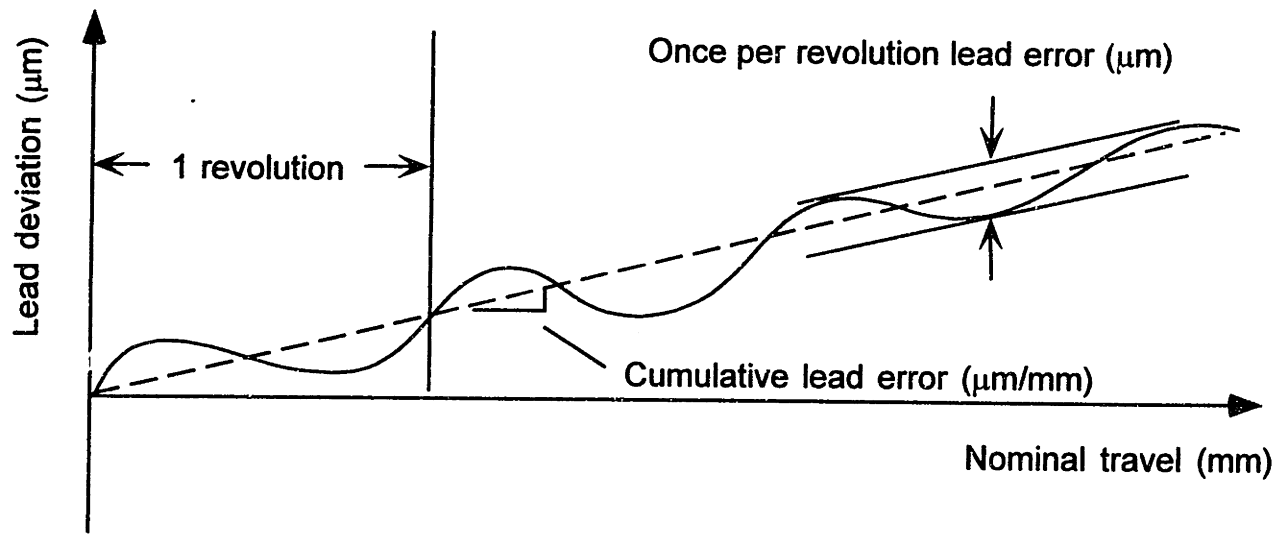


Figure 11.4.3 Cumulative and once per revolution lead error.

- Backlash (lost motion) in the x axis – Upon reversal of direction, the x axis carriage may dwell momentarily due to backlash on the ball screw.
- Deflection of the machine due to acceleration of the X axis carriage – As the x axis carriage reciprocates, the inertia of the carriage results in a periodically varying load on the structural loop of the machine.
- Misalignment (height) of the x axis carriage – The x axis carriage should produce motion along a line that intersects the work spindle axis (see Fig. 11.4.4). Any error in the height of the x axis carriage will therefore cause an error in the location of the crankpins.

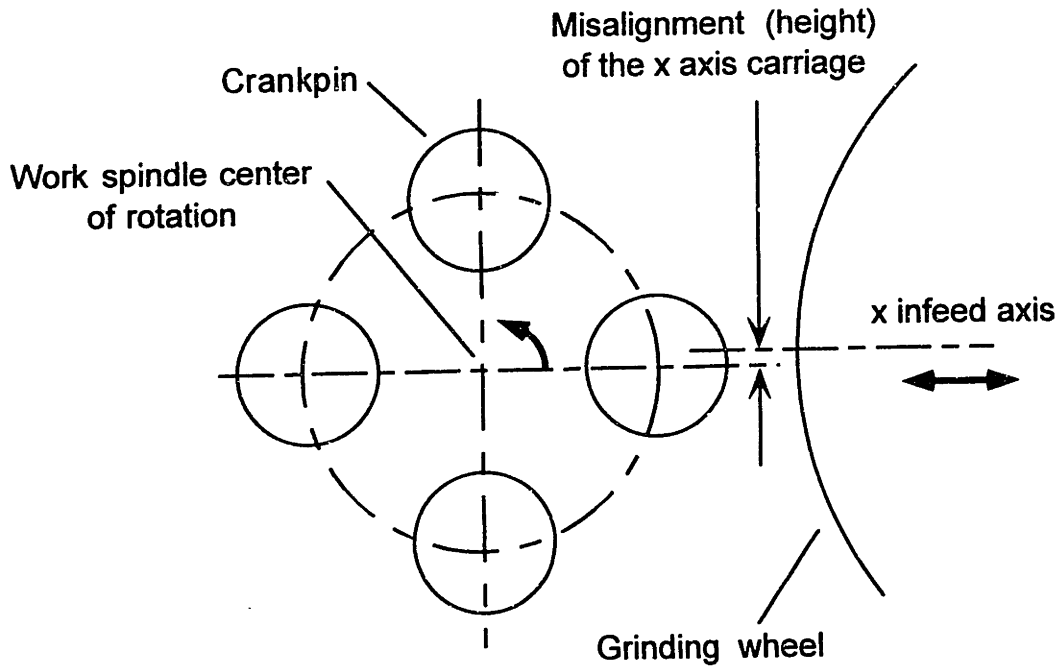


Figure 11.4.4 Misalignment (height) of the x axis carriage.

- Z squareness of the x axis carriage – The x axis infeed direction should intersect the work spindle axis at a right angle as viewed in the x-z plane. Any deviation from this squareness will cause an error in the parallelism of the crankpins with the main bearings.
- X Parallelism of the work spindles with the z axis – The work spindle axes may not be perfectly parallel with the motion of the z axis. This will tend to cause size errors varying among the crankpins and slight taper of each crankpin.

11.5 Modeling the CNC Crankpin Grinding Machine

The purpose of the modeling effort was to capture the response of the CNC crankpin grinding machine with respect to its noise factors. For this purpose, it was sufficient to assign local coordinate systems to only the main carriages, spindles, and workpiece features as shown in Figures 11.5.1 and 11.5.2. The complete kinematic model is presented in Appendix E (employing the MathCad modeling language). This section outlines some of the salient features of that model.

The base local coordinate system (LCS) location is arbitrary. In the model, the base LCS is placed on top of LCS $\mathbf{T}_2$ which simplified construction of the model. In Figures 11.5.1 and 11.5.2, the base LCS is set off to the corner for clarity.

The first LCS in the workpiece chain ($\mathbf{W}_1$) is attached to the z axis carriage at the carriage center of stiffness but vertically aligned with the ball screw. LCS $\mathbf{W}_1$ poses to accommodate the different axial positions of the six crankpins.

The next LCS in the workpiece chain ($\mathbf{W}_2$) is attached to the left workhead spindle at the spindle centerline. LCS $\mathbf{W}_2$ rotates in five degree increments to generate a full 360 degree rotation of the workhead spindle; The pose matrix for LCS $\mathbf{W}_2$ is a function of the angle of rotation a.

The LCS $\mathbf{W}_3$ is attached to the #1 (leftmost) main journal bearing on the journal bearing axis of symmetry. The final LCS ($\mathbf{W}_4$) is attached to the crankpin on the crankpin axis of symmetry and aligned so that the y axis points toward top dead center of each crankpin. Since there are six crankpins, there are six distinct $\mathbf{W}_4$ LCSs each with its own location and orientation.

The first LCS in the cutting tool chain ($\mathbf{T}_1$) is attached to the x axis carriage at the carriage center of stiffness but vertically aligned with the ball screw. The $\mathbf{T}_1$ pose matrix is a function of the x axis infeed $x(a)$ which is slaved to the work spindle angle a in accordance with the control Equation 11.3.1.

The next LCS in the cutting tool chain ($\mathbf{T}_2$) is attached to the grinding wheel spindle at the spindle centerline. The final LCS ($\mathbf{T}_3$) is attached to the grinding wheel on the wheel centerline and aligned with the center of the grinding wheel face.

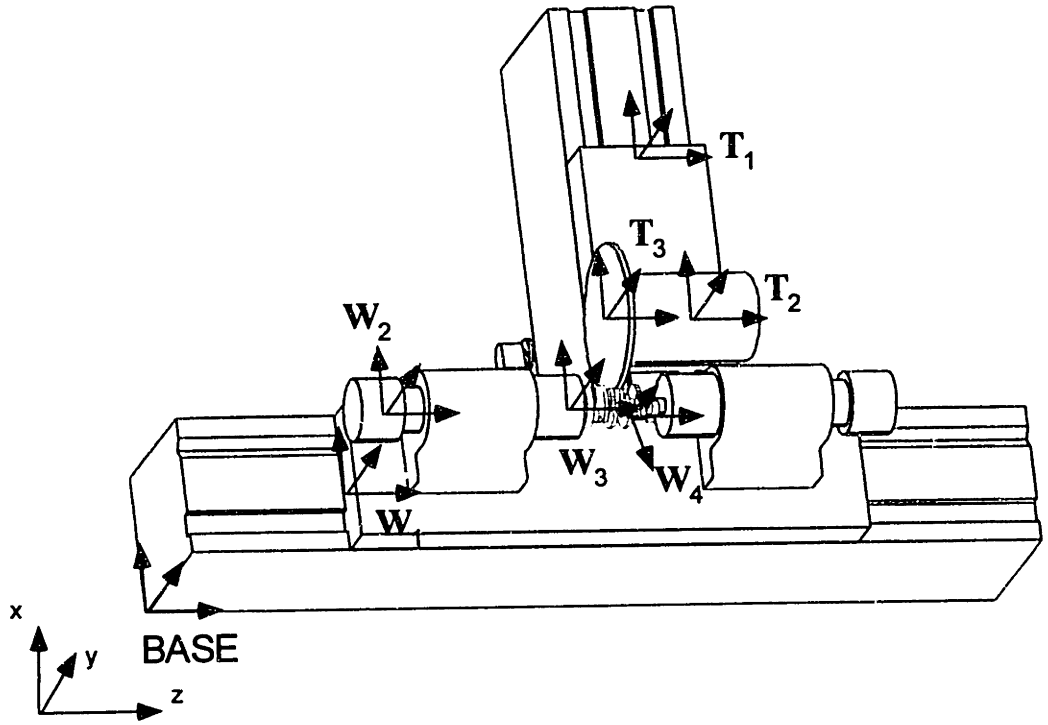


Figure 11.5.1 The CNC crankpin local coordinate system assignments on the machine schematic drawing.

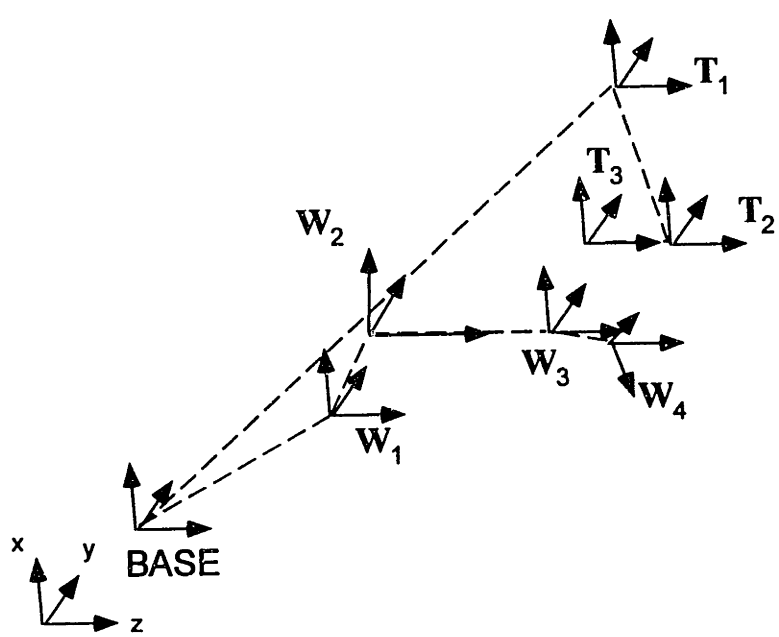


Figure 11.5.2 The CNC crankpin local coordinate system assignments with interconnections.

11.6 Verifying the Kinematic/Geometric Model

The model described in Section 11.5 is a representation of the kinematic and geometric behavior of the CNC crankpin grinding machine. It should allow one to make predictions concerning the geometry of a crankshaft based on knowledge of the noise factor values extant during its manufacture.

To provide a spot check of the model, engineers at Landis imposed known geometric errors on a CNC crankpin grinding machine. Specifically, they simulated grinding wheel attrition by increasing the grinding wheel radius in the controller and compensated for size with x axis infeed. At the same time, they simulated a main bearing size error by shimming the main bearings by a known amount in the 'V' block fixtures. They proceeded to grind all six crankpins. The crankpins were then measured on an Adcole™ model 1200 Crankshaft inspection gage (see Figure 11.6.1).

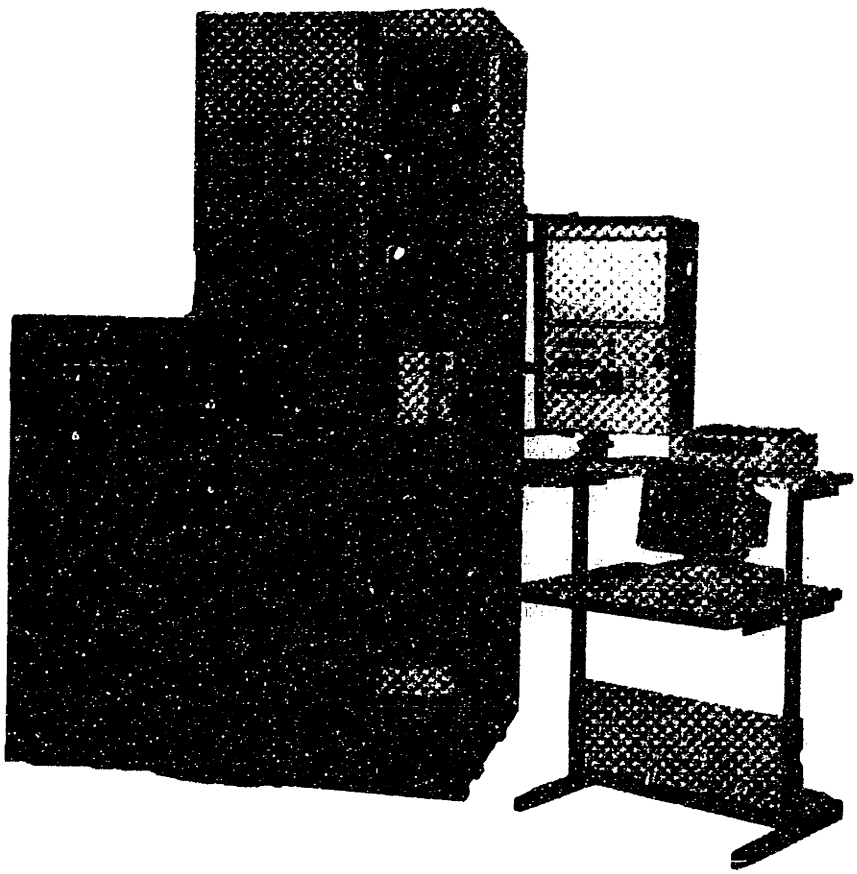


Figure 11.6.1 The Adcole™ Model 1200 crankshaft inspection gage.

The virtual machining model was used to simulate grinding a crankshaft with the same set of noise factors as existed in the cutting test. The results from the model and the cutting test are compared in Figure 11.6.2. The model predicts that the pattern of radial error will be identical in all six crankpins. Therefore, the data from all six pins are displayed in the same graph. The data and the model agree to within about one micron (which is only slightly greater than the repeatability of the measuring instrument).

Given that the errors being modeled here are strictly kinematic and geometric, it is not surprising that they can be modeled with precision. The cutting test primarily serves to ensure that no blunders were made in coding the model. It also serves as a second confirmation of the swept envelope algorithm introduced in Chapter 9.

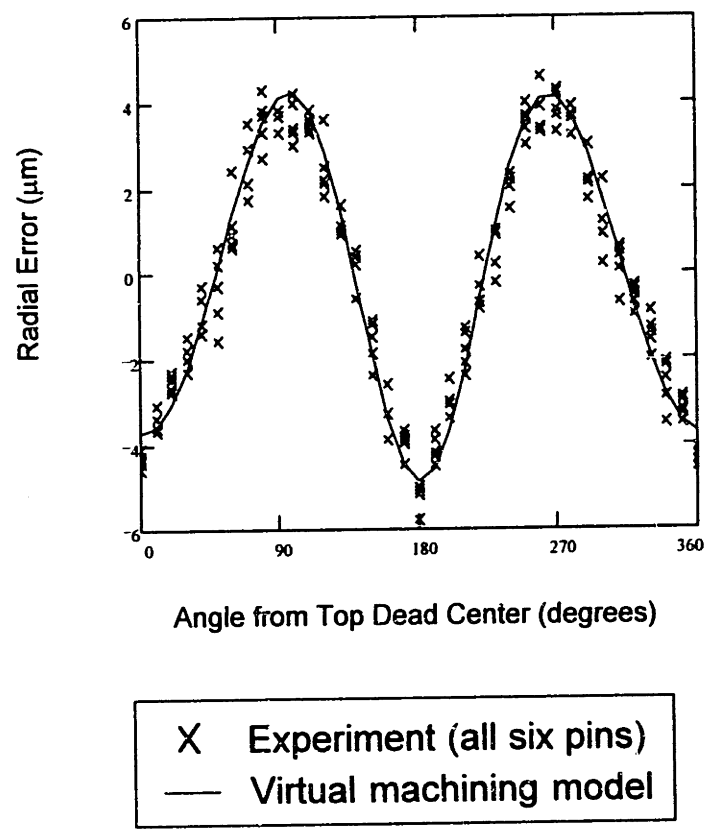


Figure 11.6.2 Comparison of cutting test data to virtual machining model predictions.

This section has reviewed a spot check of a deterministic kinematic/geometric model of the CNC crankpin grinding process using a single cutting test. In the next several sections, I use the

error signature concept to develop a statistical model of the process which is then checked against statistical process control (SPC) data.

11.7 Noise Signatures for CNC Crankpin Grinding

Noise signatures are the distinct patterns of error across the surface of the workpiece caused by individual noise factors (for more depth, see Section 9.5). The noise signature of any noise factor can be created with the virtual machining model described in Section 11.5. Near the top of the code in Appendix E, there is a list of assignments of noise factor values. Set a single noise factor to 3σ from its target while leaving the other noise factors at their target values. The code will automatically generate the noise signature matrix S. The code writes a file to disk that includes data sufficient to reconstruct a noise signature. The noise signatures for all of the noise factors listed in Section 11.4 are plotted in Appendix F.

For crankshafts finished on a CNC crankpin grinding machine, the magnitude of error is a function not only of radial and axial position of a point on the surface with respect to the crankpin, but also of the radial and axial position of the crankpin with respect to the #1 and #4 main bearings. For example, consider the plot of radial error in the number III and V crankpins due to main bearing size mismatch (see Figures 11.7.1 and 11.7.2 respectively). Clearly, some noise factors have effects that vary significantly from crankpin to crankpin. Therefore, in order to capture the complete effects of a noise factor, its noise signature must include graphs of all six crankpins.

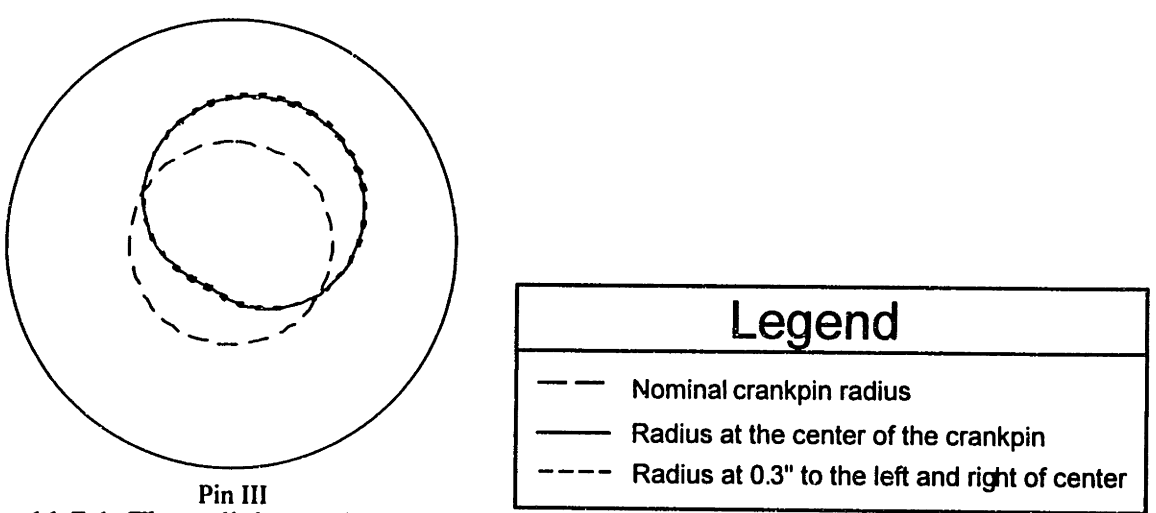


Figure 11.7.1 The radial error in the number III crankpin due to main bearing size mismatch.

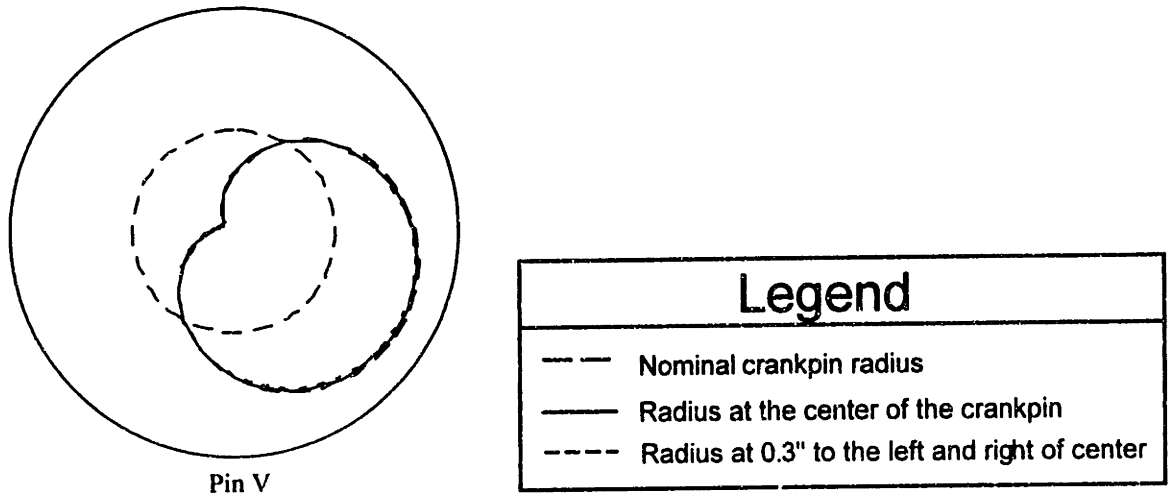


Figure 11.7.2 The radial error in the number V crankpin due to main bearing size mismatch.

For each noise factor, one may create a table of the calculated index, throw, size, roundness, and taper errors for each crankpin (for definitions of these dimensions, see Section 11.1). These values may be used to create a process capability matrix as described in Section 11.10.

11.8 Manual Diagnosis in CNC Crankpin Grinding

Using the data base of error signatures in Appendix F, it is possible for engineers and machine operators to diagnose the cause of shifts in machine performance. If the graph of crankshaft inspections data reveals a repeatable signature, then it is useful to seek a match with the one of the error signature graphs for known or suspected noise factors.

Shifts in machine performance have already been detected using the procedure described above. A CNC crankpin grinding machine (which had previously been performing well) began producing crankshafts with the error signature depicted in Figure 11.8.1. In attempting to diagnose the problem, the engineers had access to the noise signature graphs similar to those in Appendix F. A similarity was noted between the observed signature and the signature for cumulative lead error (Figure 11.8.2). This machine was configured somewhat differently than the one in the model – the machine had a scale on the x axis rather than an optical encoder on the lead screw. Therefore, the error most nearly analogous with cumulative lead error is a uniform contraction of the scale. Inspection of the scale revealed the cause of the shift in machine

performance. A clamp screw for the linear scale had loosened in service. Re-torquing the screw resulted in a reduction of roundness error to less than 1/3 the previous value as observed in Figure 11.8.3.

The results the application of virtual machining to machine fault diagnosis are promising. Noise signature graphs generated by virtual machining have been provided to machine operators on a trail basis. It has been proposed that such plots be made a standard part of the trouble shooting guides that are part of every machine's operating documentation.

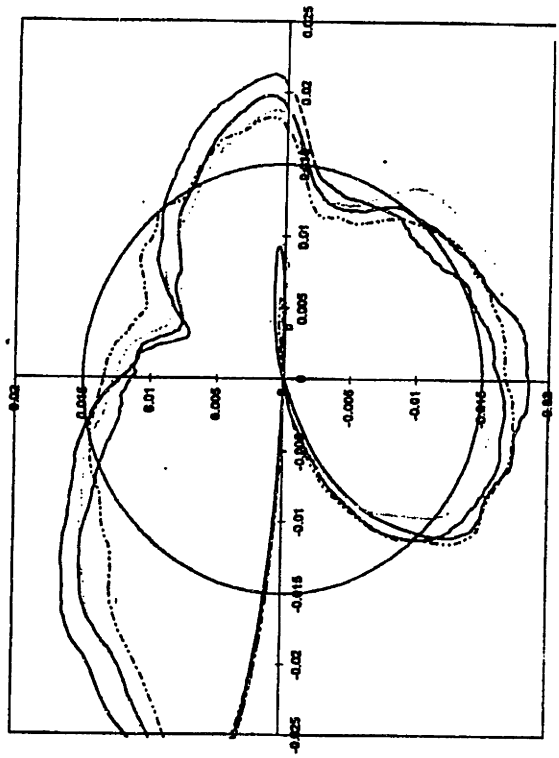
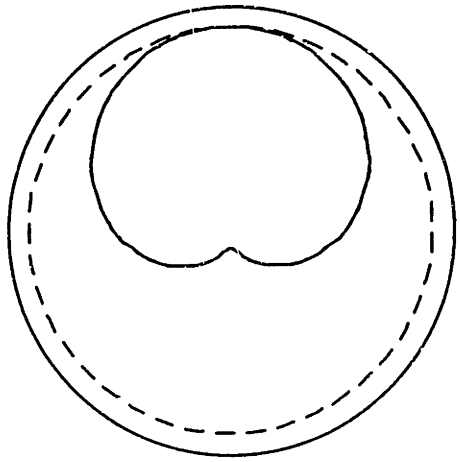


Figure 11.8.1 Post process gauging data from a 4 cylinder crankshaft revealing the effect of a loose scale on the x axis.



Pin I

Figure 11.8.2 Virtual machining output for cumulative lead error in the x axis.

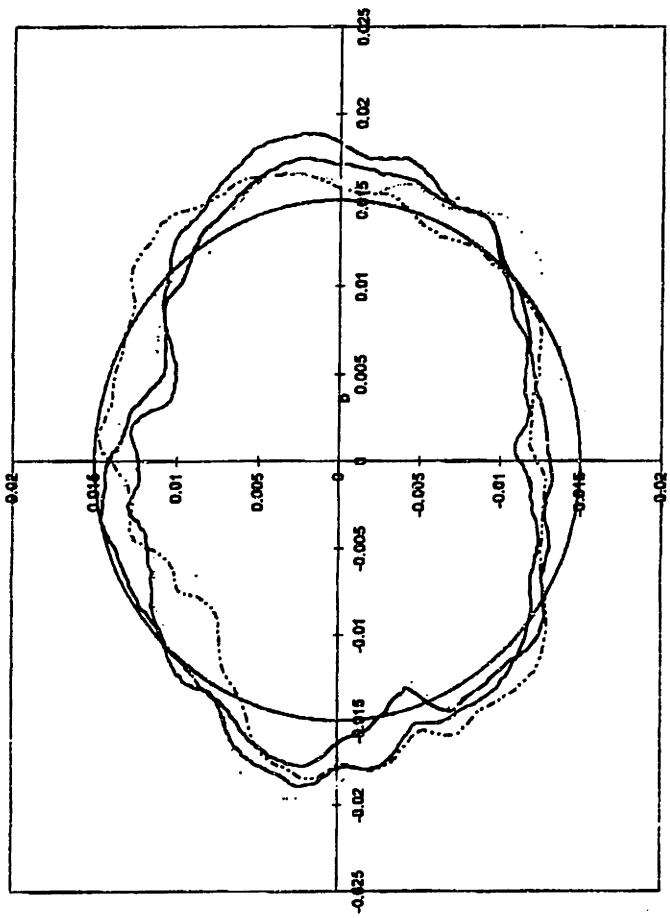


Figure 11.8.3 Post process gauging data from a 4 cylinder crankshaft. after the scale was tightened

11.9 Non-linearity and Tolerances of Form

The virtual machining model of the CNC crankpin grinding machine helps to illustrate the fundamental non-linearity of many tolerances. This phenomenon was introduced in Section 9.5.4 where I showed that tolerances (such as tolerances of position) are a linear mapping of a noise signature whereas tolerances of form (such as roundness) are a non-linear mapping.

To illustrate this principle, let us consider the effect of two of the noise factors (height of the x axis carriage and cumulative lead error) on the roundness of the number one crankpin. When I run the virtual machining model of Appendix E with the noise factor "height of the x axis carriage" set to its 3σ value (and all others set to zero), the roundness error of the number one crankpin is 1.4 μm (see Figure 9.6.3.1). When I run the virtual machining model of Appendix E with the noise factor "cumulative lead error" set to its 3σ value (and all others set to zero), the roundness error is 1.4 μm (see Figure 9.6.3.1). However, if I activate both noise factors simultaneously, the crankpin has a roundness error of 1.5 μm. The superposition principle clearly does not hold for roundness of crankpins in this case.

The non-linearity observed in Figure 9.6.3.1 requires some further explanation. If one selects any angular position on the crankpin, the radial error in the superposed case is the sum of the radial errors due to the two contributing effects. In other words, the non-linearity does not arise from the mapping from noise factors to geometry ($M(\mathbf{n})$) but from mapping geometry to roundness. This reinforces the principle behind the adapted Monte Carlo approach developed in Section 9.6.3 and demonstrated in the next section.

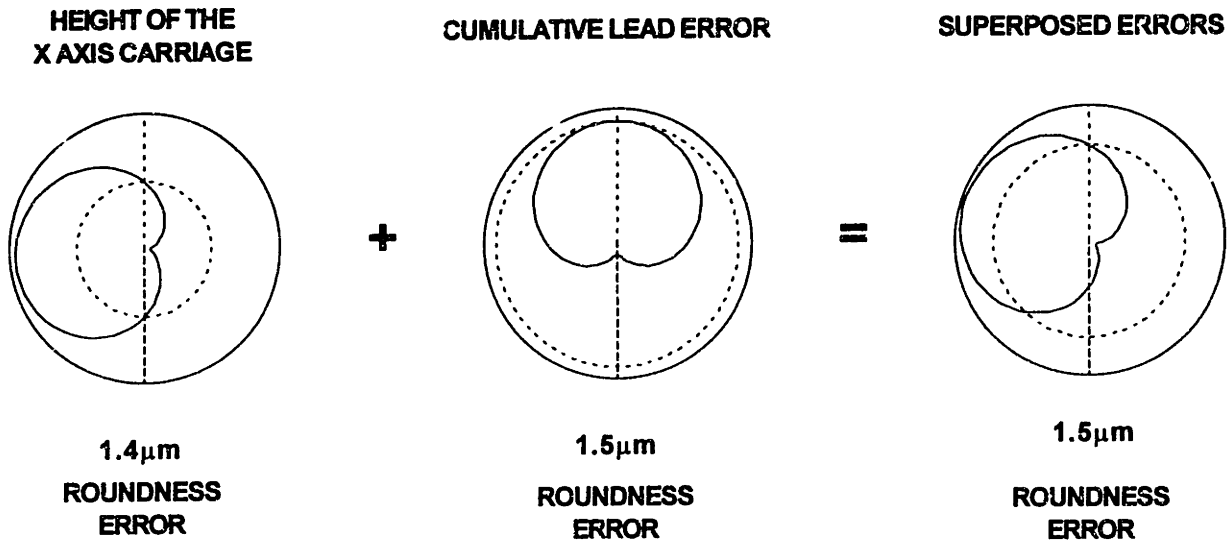


Figure 11.9.1 Superposition of tolerances of form.

11.10 Monte Carlo Error Budgeting in CNC Crankpin Grinding

The CNC crankpin grinding process has to meet all the tolerances defined in Section 11.1 in the presence of noise factors listed in Section 11.3. An error budget based on the algorithms presented in this thesis helped engineers meet the tolerances despite the noise factors; It even resulted in some cost saving changes in the assembly process specifications for CNC crankpin grinding machines.

Some details of the complete error budget are proprietary. Therefore, rather than document the budget in detail, I will mask the error budget and present a portion of it that illustrates some of the key concepts. Please note that the results and design recommendations in this example problem do not reflect the actual results and recommendations made to Landis. The data presented here was masked normalizing by arbitrary tolerance values on both noise factors and quality characteristics. This procedure retains most of the statistical regularities and patterns in the experimental data, but masks the true capabilities of the production equipment.

Let us consider a simplified error analysis problem. I will assume that only two noise factors are significant –grinding wheel attrition and cumulative lead error. Let us assume that this noise factor "grinding wheel attrition" is normally distributed. The noise factor "cumulative lead error" may not be normally distributed. The specification on cumulative lead error is set by the manufacturer of the lead screw. Typically, there is 100% inspection of these items. Only the

most accurate are sold as "class A" lead screws. Since the accuracy is "inspected in", the distribution of the "class A" lead screws is a segment of a normal distribution. If the segment is small enough, the distribution will be very nearly uniform. Let us assume we have specified "class A" lead screws and that the distribution is nearly uniform.

Using a random number generator to sample from the assumed distributions, I created 900 pairs of noise variable values and displayed them in the scatter plot of Figure 11.8.1. The 900 points represent 30 "virtual" CNC grinding machines each of which manufactures 30 virtual crankshafts. This accounts for the striped appearance of the graph. Each vertical stripe represents a single virtual machine. Each point in a stripe represents a crankshaft ground on that machine. The uniform distribution of cumulative lead error is evident in the single, wide vertical band of stripes in Figure 11.10.1.

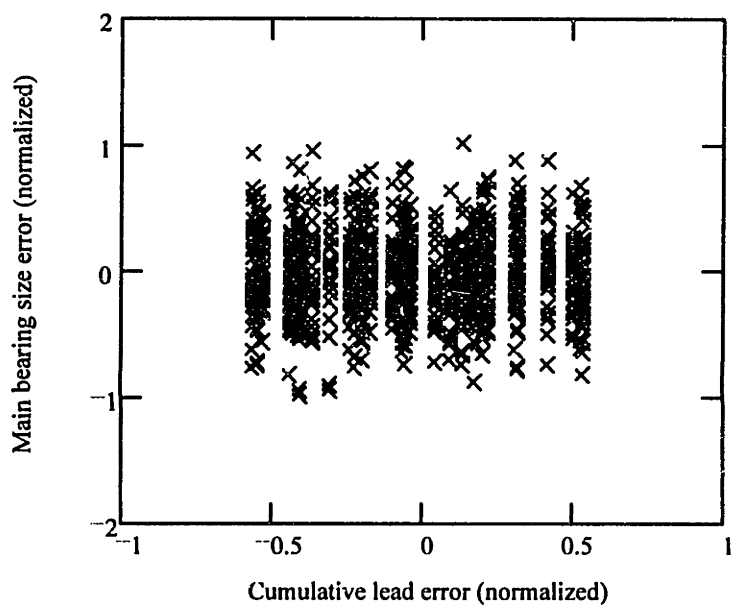


Figure 11.10.1 Noise factors in a Monte Carlo simulation.

The noise variable pairs depicted in Figure 11.10.1 were transformed using the error signature matrix to create 900 virtual crankshafts. The virtual crankshafts were then "measured"; The set of points that comprises the virtual crankshaft was transformed into roundness and throw values for the number one crankpin using Equations 9.6.3 and 9.6.4 (see Figure 11.10.2).

Figure 11.10.2 captures some of the critical behaviors of the CNC crankpin grinding process. One key observation is that the virtual measurements fall in a V shape centered on the origin. No part of the V is in the lower half plane since roundness error is positive by definition. One consequence of this V shape is that there is no significant correlation between roundness error and throw error, but there is a significant correlation between roundness error and the *absolute value* of throw error. Also, the data for individual machines is clustered in vertical bands along the V.

I was interested in confirming the ability of the Monte Carlo approach to virtual machining to identify such statistical regularities in manufacturing processes. To this end, I employed data collected by the grinding machine manufacturer. They ground 50 V8 crankshafts on one machine and 48 on another. They measured several quality characteristics of each crankpin. This data is presented in graphical form in Appendix G.

The roundness and throw value pairs for the number one crankpin of all 98 crankshafts are depicted in Figure 11.10.3. Each point represents the throw and roundness error values for the number one crankpin on a single crankshaft. The pattern emerging in the data seems to confirm the hypothesis that throw error is correlated with absolute value of roundness error; There is a statistically significant correlation at the α=0.01 level. The least squares fit to the data is depicted in the plot as a dashed line.

The data also tend to confirm the hypothesis that the output of individual machines will be grouped along the V. The horizontal scatter in the data is not predicted by the model. This should not be surprising as there are only two noise factors in this simplified model. Many more noise factors actually exist in the operation of the machine. Nonetheless, the close agreement in the patterns in the data and model tends to confirm the Pareto principle; Consideration of just a few key noise factors can provide a reasonable model of the behavior of a production process with respect to noise.

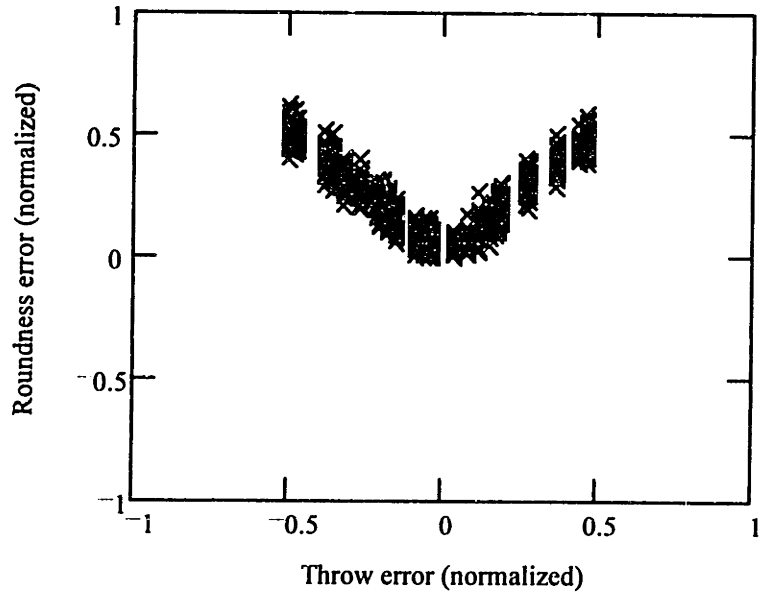


Figure 11.10.2 Quality characteristics from a CNC crankpin grinding machine – Monte Carlo simulation.

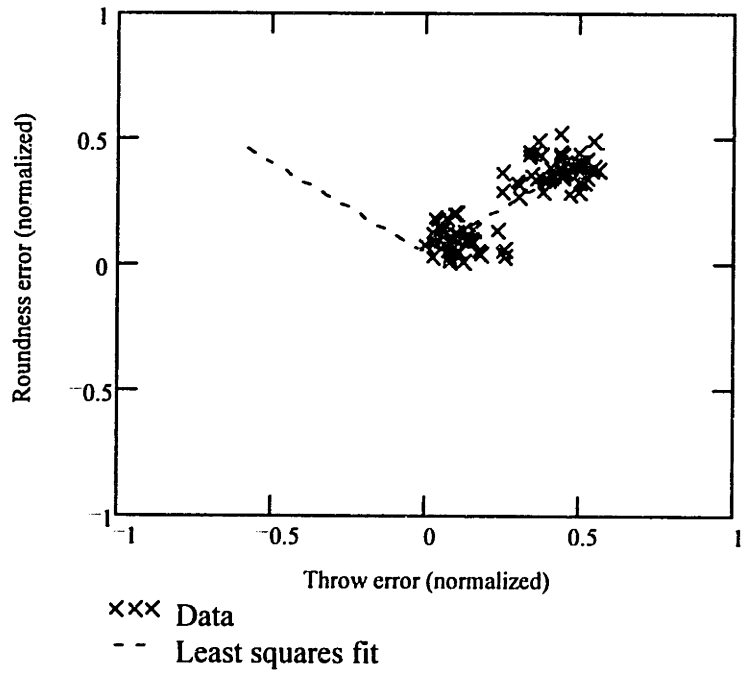


Figure 11.10.3 Quality characteristics from a CNC crankpin grinding machine – experimental data.

The simplified Monte Carlo model of the crankpin grinding process presented here is useful in error budgeting. Figure 11.10.2 reveals that there is some margin for both throw error and roundness error. It may therefore be possible to relax the specifications some of the noise variables. Cumulative lead error affects both throw and roundness. Therefore, one may infer that tolerances on cumulative lead error can be relaxed considerably while still meeting product tolerances. To test this hypothesis, I expanded the range of variation in cumulative lead error by 50% and repeated the Monte Carlo simulation. All 900 virtual crankpins from the simulation met the tolerances on roundness and throw (see Figure 11.10.4).

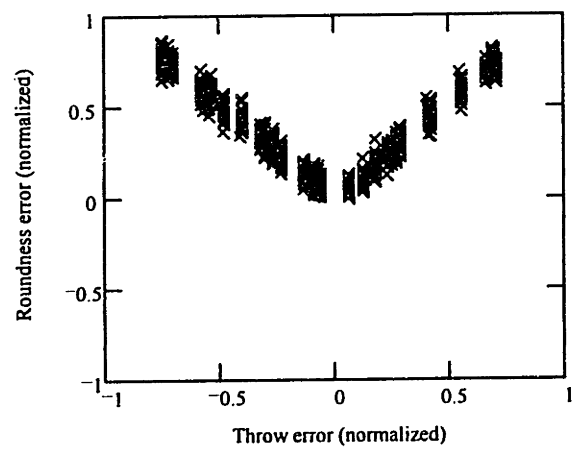


Figure 11.10.4 Quality characteristics from a CNC crankpin grinding machine – Monte Carlo simulation with relaxed specifications.

This example in this section has illustrated the use of Monte Carlo simulation in exploring the statistical behavior of a manufacturing process. It has also shown its utility in guiding design decision making (especially in tolerance design). The next section compares this approach with the process capability matrix based approach.

11.11 The Process Capability Matrix for CNC Crankpin Grinding

The simplified error budget case study in Section 11.10 reveals some of the limitations of the capability matrix approach in error budgeting. Let us call quality characteristic number one is throw and quality characteristic number two is roundness. Let us call noise factor number one is cumulative lead error and noise factor number two is main bearing size error. The capability matrix (Equation 3.9.1) for the system with relaxed tolerances described at the end of Section 11.10 is

$$\mathbf{C} = \begin{bmatrix} 0.015 & 0.80 \\ 0.8 & 0.18 \end{bmatrix} \tag{11.11.1}$$

and the bias vector is identically zero.

The Monte Carlo method applied to this capability matrix and bias vector results in a linear transformation of the noise factors (whose distribution is depicted in Figure 11.10.1). In this case, the capability matrix model provides the same information as the more general Monte Carlo approach in Section 11.10. There appears to be about the same margin for relaxing specs in Figure 11.11.1 as there is in 11.10.2. In this case, the destructive interference discussed in Section 9.5.4 and 11.7 did not significantly interfere with the effectiveness of a capability matrix approach to error budgeting.

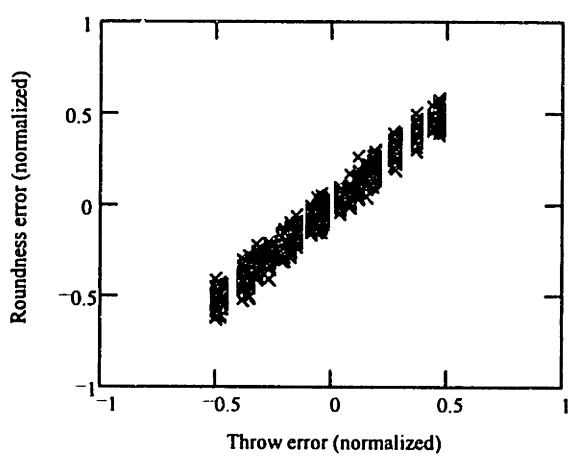


Figure 11.11.1 Quality characteristics from a CNC crankpin grinding machine – Monte Carlo simulation over a linearized (capability matrix) model.

The upper and lower bounds for rolled throughput yield (Equation 5.6.4) can be computed for the capability matrix (Equation 11.11.1). This equation does not strictly apply to this case study since one of the variables is not normally distributed nor is the behavior linear. Still, the equation predicts a reasonable yield range of $99.979\% \leq Y_{RT} \leq 99.98\%$ -- zero defects for practical purposes in this context. By comparison, the Monte Carlo simulation had zero defects out of 900 crankshafts.

In some respects, the capability matrix approach can provide misleading results. The non-linear model has a mean normalized roundness error of 0.27 and a standard deviation of 0.16. The linearized (process capability matrix) model has a mean normalized roundness error of near zero and a standard deviation of 0.31. It is clear that the differences are significant. The negative roundness errors generated by the capability matrix model (which make no physical sense) tend to throw off the predicted mean and standard deviation of roundness. One should interpret results of capability matrix calculations for non-linear tolerances with caution (see Table 9.5.4.1 to determine which tolerance have non-linear behavior).

11.12 Chapter Conclusions

The CNC crankpin grinding process is a revealing case study of both the virtual machining and capability matrix concepts. The machine clearly reveals the complex interactions and correlations among noise factors and quality characteristics. Some of the key points in this case study that reinforce earlier theoretical developments are:

12. Related Work

This goals of this thesis are closely related to several established areas: six sigma producibilty analysis, Taguchi methods, response surface methodology, error budgeting, axiomatic design, and tolerance analysis. Each section in this chapter briefly reviews each of these topics and discusses their bearing on this thesis.

12.1 Six Sigma Producibility Analysis

12.1.1 Review of Six Sigma Producibility Analysis

Six sigma producibility analysis is an approach to manufacturing system analysis and design developed primarily at Motorola in the 1980's [Harry and Lawson, 1992]. It is a kit of probabilistic and statistical tools especially suited to issues of manufacture. It is especially well suited to characterization and tuning of existing processes. This section will review the key elements of six sigma methodology.

Before proceeding with a review of six sigma methodology, it is worthwhile to clarify a common confusion about its name. Six sigma refers to a "stretch goal" that is often associated with the methodology. The stretch goal is to achieve a process with low enough variation that the half tolerance width equals six standard deviations. If bias were zero, then such six sigma capability would result in 2 defects per billion units. If bias is 1.5 standard deviations (as is often assumed to be the case), then there will be about 4 defects per million units. The stretch goal and the methods proposed to achieve it should be clearly separated. One may question the goal without disparaging the methods and vice versa.

A Motorola licensed course [Hughes Electronics Corporation, 1994] defines the following "six steps to six sigma":

1) Define requirements
2) Identify the key traits to meet requirements
3) Analyze variables that drive quality
4) Establish targets and tolerance
5) Measure performance

6) Adjust and control the design and process

The first four steps are important, but are not the central contribution of six sigma methodology. Steps one through four are in the domain of Total Quality Management [Shiba, et. al.], Total Design [Pugh, 1990] and other product development approaches. The sixth step is also given little attention in six sigma methodology being the domain of Response Surface Methodology and Robust Design. About 90% of Harry and Lawson's text [1992] is focused on the fifth step, measuring process capability. This review will maintain this focus.

To perform step five, one needs metrics of system performance. Six sigma methodologies include many such measures. The capability index C_p and performance index C_{pk} are defined in section 1.3.1 and are commonly employed in six sigma methodologies. Defects per unit *dpu* is the expected value of the number of non-conforming quality characteristics in any discrete unit of product. First time yield Y_{FT} is the probability that a product will meet any given requirement. Rolled throughput yield Y_{RT} is the probability that a product will pass all of its requirements.

Rolled throughput yield is recognized under six sigma approaches as the final measure of process performance. It determines how much scrap and rework must be performed, the amount of inspection required, and ultimately has a substantial effect on the profitability of the enterprise. In 1987, Motorola estimated that the cost of non-conformance was as much as 15% of total sales. This is typical of a process operating at 4σ level of performance (C_p=1.33). By moving to 6σ performance, a company will lose less than 1% of revenue due to non-conformance [Hughes Electronics, 1995] .

Rolled throughput yield may be the ultimate measure of product/process performance, but it can be difficult to measure directly. If a process has reasonable performance, then large numbers of products must be manufactured and inspected to directly measure yield with a reasonable degree of confidence. For example, a process with Cpk of 1.33 will have a dpu of only one part in 15,000. It may be impracticable to manufacture and inspect that many parts just to characterize the process. Therefore, six sigma methodology provides several means to compute Y_{RT} based on other performance metrics.

Harry and Lawson [1992] suggest that, in most cases, it is acceptable to assume that the quality characteristics are independent which leads to the following simple formula

$$Y_{RT} = \prod_{i=1}^{m} Y_{FTi} \tag{12.1.1.1}$$

Equation 12.1.1.1 provides a simple way to estimate the probability of meeting all the requirements given many opportunities for non-compliance.

Harry and Lawson [1992] claim that if the probability of non-conformance per unit is less than ten percent and the overall likelihood of observing a defect is high, then a Poisson model can be used to model the process. The Poisson model is

$$P(r) = \frac{dpu^r e^{-dpu}}{r!} \tag{12.1.1.2}$$

where $P(r)$ is the probability of observing exactly r defects in a discrete unit of product and dpu is the number of defects per unit of product. Given the definition of rolled throughput yield as the probability that there are zero defects in a discrete unit of products, it follows that

$$Y_{RT} = e^{-dpu} \ . \tag{12.1.1.3}$$

The above formulae (12.1.1-3) are an essential part of six sigma methodology. Rolled throughput yield is a key metric determining profitability. Accurate estimation of yield is therefore the key to assessing the effects of variation on process performance. Yet these formulae assume independence among product quality characteristics. The effects of this assumption will be explored in the following section.

12.1.1 Bearing on this Thesis

The goal of this thesis is to help engineers design manufacturing systems whose products conform to their dimensional tolerances. To do this, it is necessary to evaluate the capability of proposed designs to meet these tolerances. Therefore, this thesis shares many of the goals of six sigma producibility analysis and can be considered an extension of the field. The key difference is that this thesis permits consideration of correlation among quality characteristics which may have a substantial effect on estimates of process capability.

Six sigma producibility analysis provides formula to estimate rolled throughput yield from other producibility metrics (Equations 12.1.1.1 through 3). These formulas all assume probabilistic independence among quality characteristics. In the case of Equation 12.1.1.1, the methodology explicitly acknowledges the limitation of the formula but claims that reasonable

estimates may be obtained in most cases of practical interest. This thesis demonstrated that for the MCM surface mount process, Equation 12.1.1.1 gave estimates two orders of magnitude in error. In the CNC crankpin grinding process, the formula was in error by over 50%.

The Poisson model of Equations 12.1.1.2 and 3 is not accompanied with an appropriate caveat in Harry and Lawson [1992]. However, these formulae are also restricted to the case of independence among quality characteristics. The Poisson process is a probabilistic description of the behavior of arrivals at points on a continuous line. Let $P(k,t)$ represent the probability that there are exactly k arrivals in an interval of duration t. The definition of a Poisson process is given by the following two properties [Drake, 1967]:

1) For any suitably small value of Δt, $P(k,\Delta t) = \begin{cases} 1-\lambda\Delta t & k=0 \\ \lambda\Delta t & \mathrm{k}=1 \\ 0 & \mathrm{k}>1 \end{cases}$

2) Any events defined on non-overlapping time intervals are mutually independent.

Based on this definition, one can solve a simple differential equation to show that the Poisson probability mass function is

$$P(k,t) = \frac{(\lambda t)^k e^{-\lambda t}}{k!} \quad t \geq 0, \;\; k = 0,\, 1,\, 2,\, \ldots \tag{12.1.2.1}$$

With this PMF, one can show that λ is the inverse of the expected value of the inter-arrival time and λt is the expected value of number of arrivals in any given length of time t.

Making the analogy with the Poisson model of 12.1.1.2, a *defect* corresponds to an *arrival* and a *unit* (which will have a fixed number of opportunities for defects) corresponds to a length of *time*. Thus, the expected value of defects per unit (*dpu*) corresponds with the expected value of arrivals in any given length of time (λt). Similarly, the number of defects in the product r corresponds to the number of arrivals k. Continuing this analogy, the term "non-overlapping time interval" in property #2 above corresponds to any collection of distinct opportunities for failure. Therefore property #2 in the definition of the Poisson process implies that the first time yields must be probabilistically independent since first time yields are events defined over a collection of distinct opportunities for failure (albeit the smallest possible collection). This

reasoning shows that the Poisson model of a manufacturing process implicitly assumes independence among quality characteristics.

None of the formulae or models within six sigma methodology work well if there is substantial correlation among quality characteristics. One of the primary contributions of this thesis is that it provides methods for estimating rolled throughput yield even in the presence of correlation.

12.2 Response Surface Methodology

12.2.1 Review of Response Surface Methodology

Response Surface Methodology (RSM) is a set of methods of experimental strategy, mathematical modeling, and statistical inference for use in efficient empirical exploration of a system. In most cases, the methods are used when a *system response* (an output of a system which often represents its primary function) is affected by a large number of *natural variables*. For example, let us say that a chemical engineer is interested in the purity, η, of the product of some chemical reaction. Let us further assume that this purity is a function the natural variables: reaction pressure, ξ_1, reaction temperature, ξ_2,and reactant concentration, ξ_3, etc.. RSM assumes that there exists some functional relationship of the system response to the natural variables

$$\eta = f(\xi_1, \xi_2, \xi_3 \ldots) \tag{12.2.1}$$

RSM depends upon the approximation of the response function f by a low order polynomial in some region of the independent variables. The response function is often reformulated as a function of a vector of design variables $\mathbf{x}$ which are linear combinations of the natural variables. One possible model of a system is a second order approximating function

$$\eta = b_0 + \mathbf{x}^T\mathbf{b} + \mathbf{x}^T\mathbf{B}\mathbf{x} \tag{12.2.2}$$

One of the goals of RSM is to make an efficient experimental determination of the model parameters (b_0, $\mathbf{b}$, and $\mathbf{B}$ in 12.5.2). For this purpose, RSM employs the methods of Design of Experiments DOE developed by Sir Ronald Fisher and Frank Yates [Fisher, 1935]. First and second order models can be estimated using factorial designs, fractional factorial designs, or

simplex designs [Box, 1952]. In all these approaches, it is assumed that during the experiments the x_i can be controlled and measured with negligible error.

One of the primary uses of RSM is in experimental determination of optimum conditions [Box and Wilson, 1951]. For example, the chemical engineer mentioned earlier may wish to find a set of process variable settings which maximize the purity of the product. This can be done by finding the stationary points of the model. Taking the derivative of the model and setting it equal to zero gives the formula for a stationary point

$$\mathbf{x}_0 = -\mathbf{B}^{-1}\mathbf{b}/2 \tag{12.2.3}$$

This stationary point may be a minimum, a maximum, or a saddle point. One concern is that the stationary point may lie outside of the region over which the experiment was conducted. It is often inadvisable to extrapolate beyond that region. Therefore, RSM includes a number of techniques for searching for the region of maximum response, such as steepest ascent procedures. These methods help the experimenter to quickly explore the response surface and find a region over which a more thorough experiment may be conducted to identify a maximum.

Response surface methods were adapted to modeling and optimizing spatial uniformity in manufacture by Guo and Sachs [1993]. In some processes, one is concerned with uniformity of a product's quality characteristics at different locations within a batch. One measure of spatial uniformity within a single batch is the standard deviation of the quality characteristics at the selected points in the batch. The standard deviation happens to be a highly non-linear function of the point quality characteristics. Guo and Sachs created low order polynomial models of each point quality characteristic separately and used the definition of spatial uniformity to map the quality characteristics at the many points into the spatial uniformity metric. They found this procedure, which they called *multiple response surface methodology*, was much more efficient than fitting a high order polynomial to the uniformity data directly.

12.2.2 Bearing on this Thesis

Response Surface Methodology (RSM) is a valuable set of tools for empirical evaluation of system response. This thesis has emphasized modeling rather than experimentation for this purpose. This is primarily because the three case studies were amenable to accurate, and

reasonably simple modeling. In cases studies for which no models are available or modeling efforts are costly compared tc experiments, RSM may be employed to determine the Jacobian needed for constructing a process capability matrix. Any good text on RSM will include a section on designs for fitting first order models [e.g., Meyers, 1971].

Response surface methodologies are, for the most part, limited to single response systems. The focus of this thesis is multi-output systems with two sided constraints on the outputs. In effect, the methods of this thesis (especially of section 5) take a system with multiple responses (quality characteristics) and turn it into a system with only one response (rolled throughput yield). Viewed in this light, this thesis can be seen as a necessary pre-processor for input to an RSM approach.

This thesis has been concerned with estimating the yield of a manufacturing system at a given set of process set points. If a process engineer is interested in searching the space of set points to optimize yield, she might employ RSM as s search engine while using capability matrices and the associated formulas and algorithms as a fitness function.

The multiple response surface methodology of Guo and Sachs [1993] is very similar to the form tolerance modeling approach in this thesis. Guo and Sachs sample quality characteristics at many points within a batch; I sampled geometric error at many points across the surface of the workpiece. Guo and Sachs made the standard deviation calculation explicit in their model of the process and therefore could employ lower order polynomials of the response of quality characteristics to noise; I made the form tolerance computations explicit in my system models and therefore could employ linear models of the geometry of a part to noise.

12.3 Taguchi Methods

12.3 Review of Taguchi Methods

Beginning in 1949, Genichi Taguchi began to apply methods of statistical design of experiments (first developed by R. A. Fisher) to research and development efforts at Nippon Telephone and Telegraph. It was here that the foundations of Dr. Taguchi's methods of "Quality Engineering" began to take form. Over time, Dr. Taguchi developed a set of tools for product design, off-line quality control, and experimental design [Taguchi, 1987]. This group of tools has become known as "Robust Design" or as "Taguchi methods".

In the early 1980's, Taguchi Methods were adopted by a few American companies most notably AT&T Bell Labs, Ford Motor Company, and Xerox Corporation. Dr. Don Clausing has taken a leading role in promulgating the methods in Western industry. The methods are now widely practiced in industry and are an active area for research.

Some of the essential ideas to be reviewed in this section are P-diagrams, quality loss functions, signal-to-noise ratios, parameter design, and designed experiments.

In order to represent a process, Taguchi employs P-diagrams (see Figure 12.3.1.1). The product and process being concurrently developed are represented as a black box function. The output of the function is the response – the characteristics of the product valued by the customer. The function takes three types of inputs. The signal factor defines the desired response. The noise factors are uncontrolled variables that affect the system response. The control factors are variables that may be set by the designer to reduce the sensitivity to noise.

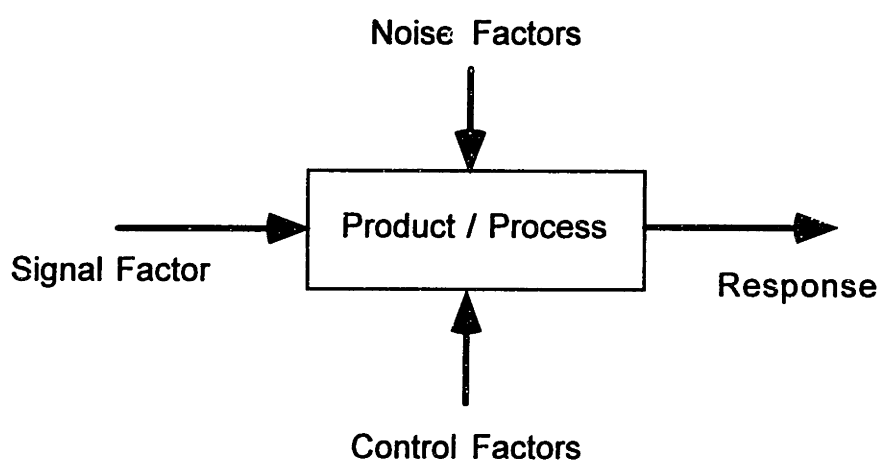


Figure 12.3.1.1 A P-Diagram for a product/process design

The response of the product and process as labeled on the P-diagram determines the quality of a product as perceived by the customer. As defined by traditional engineering specifications, either the response is good enough, or it is unacceptable. Taguchi has encouraged engineers to rethink this "goal post" mentality. He correctly points out that the distinction between a response slightly outside the tolerance band and a response slightly within the band is arbitrary. If we

think of the value we assign to a response as a function of its dimensions, engineering tolerances effectively assign a discontinuous function to this value (Fig.12.3.1.2)

As an alternative to the "goal post" mentality, Taguchi proposes a quadratic quality loss function. For plus and minus tolerances, it is defined as

$$L(q) = \frac{A_o}{[(U-L)/2]^2}\left(q - \frac{U+L}{2}\right)^2 . \qquad (12.3.1.1)$$

where A_0 is the cost to scrap or rework the product, q is the dimension or other quality characteristic of the product, and U and L are the upper and lower tolerance limits. If there are multiple system responses, one may define an average quality loss as

$$Q = \frac{1}{m}\sum_{i=1}^{m} L_i(q_i) \qquad (12.3.1.2)$$

The quadratic quality loss function is a simple, continuous function that meets the boundary conditions that quality loss must be zero when the dimension equals the basic dimension and equals A_0 at the upper and lower tolerance limits. Taguchi's quality loss function encourages decision makers to recognize that products that are closer to the basic dimensions usually perform better and create more value for the customer.

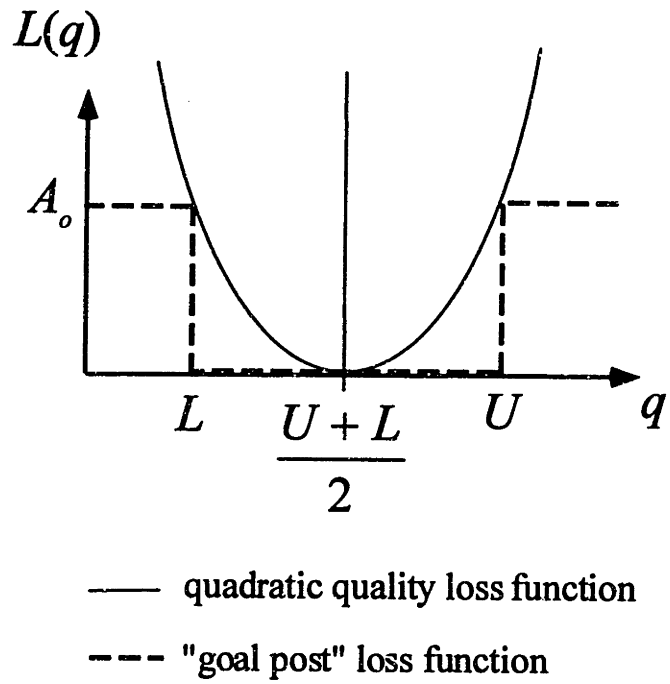


Figure 12.3.1.2 A quadratic quality loss function compared to a "goal post" loss function.

Another measure of system performance employed in Robust Design is signal-to-noise ratio. The signal-to-noise ratio is designed to provide an additive measure of variation in quality characteristics. It is defined as

$$\eta = 20\log_{10}\left(\frac{\mu}{\sigma}\right) \tag{12.3.1.3}$$

where η is the signal-to-noise ratio in decibels, μ is the mean value of the system response, and σ is the standard deviation of the system response.

In a wider view, there are three major steps in Robust Design: concept design, parameter design, and tolerance design. In concept design, the basic configuration of the product and process are set down. In parameter design, a single point is chosen in the space of possible designs defined in the conceptual phase. In tolerance design, limits are set on noise factors to ensure that a balance is struck between cost of reducing noise and quality loss caused by the noise. Most of Taguchi's methods concern parameter design.

The key to parameter design is that the set of control factors that result in "on-target" performance is generally not unique, especially if there are more control factors than responses.

The process engineer usually has some freedom in choosing the control factor settings while matching the response to the signal factor. Further, the choice of control factor settings can have a significant effect on the variance of the system response. This is due to non-linearities in the system response surface over the range of control factor settings. This concept is illustrated in Figure 12.3.1.3.

Let us assume that a single response is affected by a single input variable. Let us further assume that the mean of the input may be controlled, but that there exists a fixed variance about that mean. Therefore, the mean of the input variable may be viewed as a control factor while its variance may be viewed as a noise factor. As shown in Figure 12.3.1.3, a change in the setting of the mean to a "flatter" region of the response surface will reduce the variance in the response even in the presence of the same variance in the noise factor. If some other control factor can be used to put the response on target without undoing the gain in robustness, then process improvement will have been achieved through good parameter design.

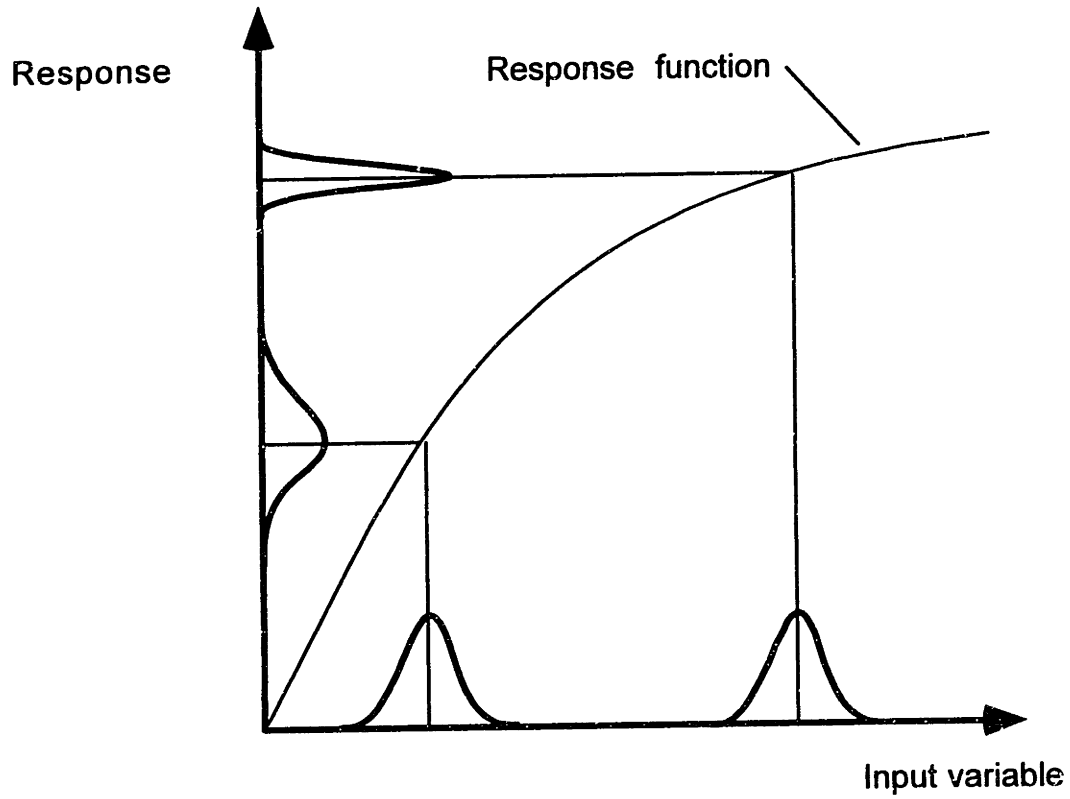


Figure 12.3.1.3 Exploiting non-linearity in system response to gain robustness to noise.

One of Taguchi's primary contributions was the application of designed experiments to robust parameter design. Experimental design concerns the assignment of treatments to experimental units so as to efficiently estimate the parameters of a low order polynomial model of a system. Taguchi methods favor fractional factorial experimental designs. Taguchi employs only a small subset of the tools available in classical design of experiments.

Robust design is an active area of research. Otto and Antonsson (1993) explored robust design including the effects of on-line quality control adjustments. Kazmer, et. al. (1996) used Monte Carlo simulation in searching for robust configurations. Many other recent contributions fall outside the purview of this thesis.

12.3.2 Bearing on this Thesis

Many of the concepts of robust design have been employed in this thesis. However, because the focus of this thesis is manufacturing system design, many of the conclusions reached are very different from those found in Robust Design.

The P-diagrams in robust design are a useful qualitative tool for categorizing the factors affecting a product and process. However, they are not a quantitative system model. Taguchi methods rely on experimental investigation for this purpose and never incorporate the experimental results back into the P-diagram. My purpose in developing manufacturing block diagrams was to create a quantitative block diagram model with associated theorems and reduction rules that aid in design decision making.

Block diagram models are suited to a different purpose than P-diagrams, but there are some similarities. In fact, with a few manipulations, P-diagrams can be transformed into manufacturing block diagrams (see Figure 12.3.2.1). First of all, this thesis is concerned with the design of the process only, not the product. As discussed in Section 1.2.1, most processes may be viewed as multi-output systems; The P-Diagram should therefore be amended to reflect the fact that there are multiple signal factors and multiple responses. The fact that the problem is multivariate introduces a significant measure of complexity. To simplify the problem, I assume for the moment that the control factors have been selected and frozen. The signal factors may be frozen since they are clearly defined on the engineering drawing of the product. To maintain the

focus on variation, the signal factors may be subtracted from the responses so that the errors may be viewed as the output of the system. Also, I divide the system into random and systematic components. These manipulations lead to the system model in Figure 12.3.2.1. If the process response is linearized and the inputs and outputs are appropriately normalized, the system may be transformed into the canonical form of the manufacturing block diagram introduced in Section 4.2 and presented for comparison in Figure 12.3.2.2.

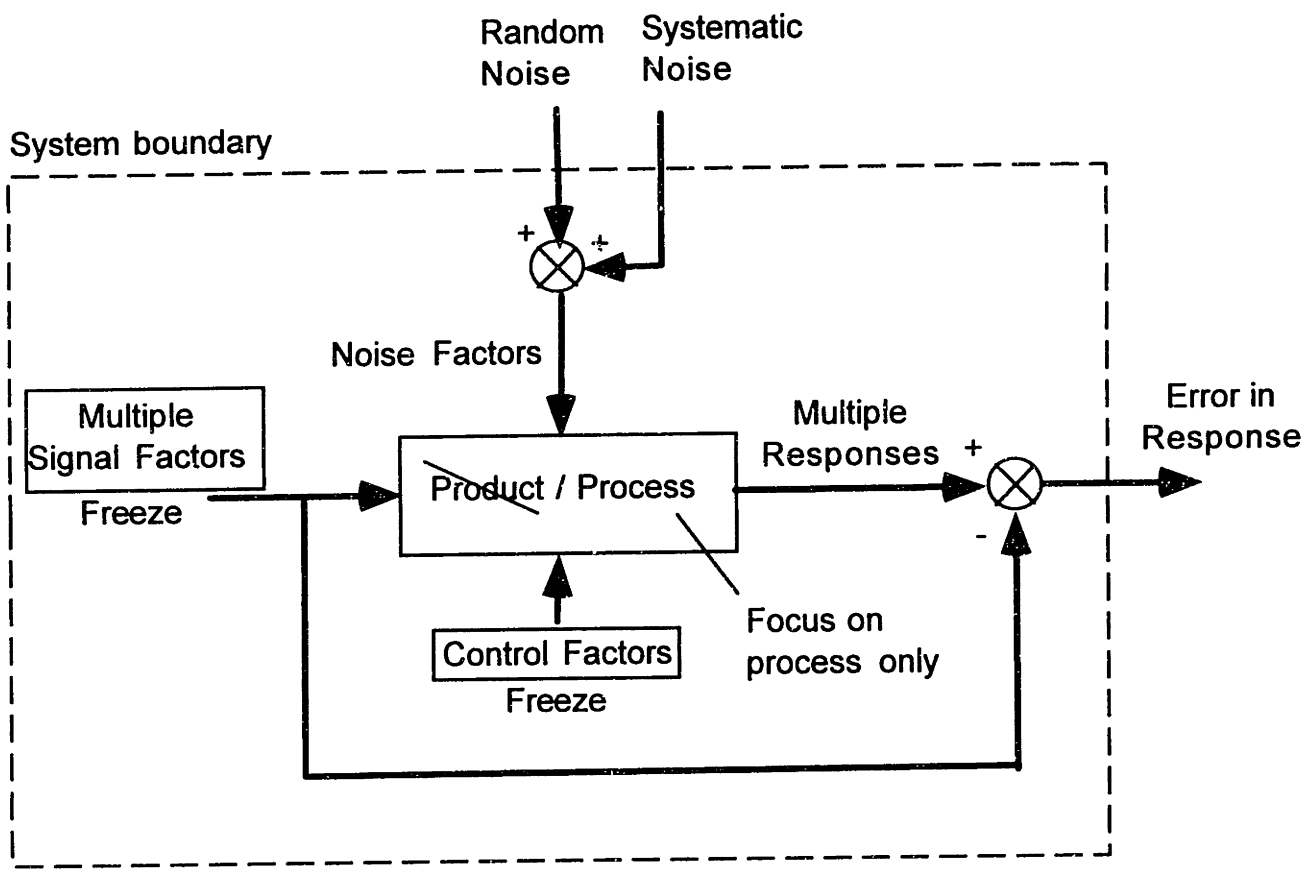


Figure 12.3.2.1 The relationship of P diagrams to manufacturing system block diagrams.

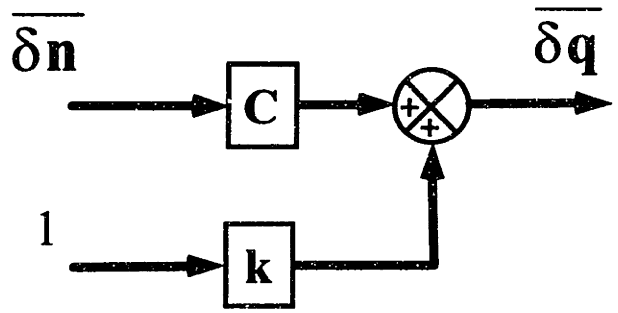


Figure 12.3.2.2 The canonical block diagram representation of a manufacturing system.

One important area of difference between Robust Design and the capability matrix approach is the choice of system performance metrics. Texts on Robust Design caution against using yield as a system performance metric because yield is not monotonic with respect to the control factors [Phadke, 1989]. Despite the mathematical inconvenience, in many cases, yield determines the profitability of a manufacturing enterprise. Therefore, ways to compute yield are valuable in design decision making. By contrast, both average quality loss and signal to noise ratios ban be deceptive indicators of system performance.

Let us consider a manufacturing system that consistently fails to meet one tolerance on a product, but meets the 99 other tolerances of the product with high fidelity. Average quality loss of this process as defined by Taguchi might be very low due to the $1/m$ term in Equation 12.3.1.2. Yet the product would have very high rates of scrap and rework due to dominance as discussed in Section 1.2.1. Average quality loss does not reflect the realities of engineering specifications as defined by ANSI.

Signal to noise ratio is also inappropriate for application to multi-output problems. The additivity of the measure fails to account for the issue of dominance. Further, the normalization with respect to mean is problematic when applied to production problems. The mean value of some quality characteristics, such as taper of a crankpin is likely to be zero or near zero. In these cases, a well centered process will be penalized with a very low signal to noise ratio.

Taguchi's applications of design of experiments to robust design can be considered as complimentary to the capability matrix approach presented in this thesis. The capability matrix approach is more strongly model based; It is appropriate for the design stage prior to prototype

fabrication if the design is amenable to first order modeling. Taguchi's approach is strongly empirical; It assumes that a production system exists which may be experimented with and tuned. I propose that the two approaches might be used in concert. The capability matrix approach might be used to evaluate feasibility in the conceptual stage of design. Later, when a prototype system is developed, Taguchi robust parameter design might be used to find robust set points. Results of the designed experiments might then be used to construct an experimentally derived capability matrix and the equations of chapter 5 might then be used to perform tolerance design.

12.4 Error Budgeting

12.4.1 Review of Error Budgeting

The design of a machine tool is driven by several factors including cost and required accuracy. The designer must be able to make a reasonable estimate of the accuracy of a machine early in the design cycle in order to make decisions on configuration, component selection, materials, and manufacturing processes (e.g. finishing processes for the ways). To make such an estimate, a detailed error budget is a useful tool. An error budget is a systematic account of all of the sources of error in a machine including such effects as component accuracy, structural compliance, and thermally induced deflections.

Error budgeting was first comprehensively applied to machine tools by Donaldson [1980]. The methods have been applied to machine tools and measuring equipment by many authors since that time [e.g., Portman, 1980, Donaldson, 1986, Treib, 1987, Narawa, 1989]. They have also been improved and extended by several researchers [e.g., Slocum, 1992, Soons, et. al., 1992, Frey, 1997]. The salient concepts in any error budgeting method are Abbé error, kinematic modeling, sensitive directions, and error combination rules.

Abbé error is the amplification of small angular errors due to large lever arms. The effect takes its name from Ernst Abbé who first noted the its importance in measurement systems. Abbé expounded the principle "If errors in parallax are to be avoided, the measuring system must be placed co-axially with the axis along which displacement is to be measured on the workpiece." Bryan [1989] extended the notion to machine design. To make a machining

operations more accurate, the workpiece should be placed co-axially with the center of stiffness of the bearings of a carriage.

Kinematic models of a machine tools can be viewed as an automatic accounting system for Abbé effects. The value of kinematic models in machine tool error budgets has been recognized by Donaldson [1980] and Slocum [1997]. Many different frameworks for kinematic modeling of machine tools have been proposed. Some researchers use a formalism based on screw theory [Ziegert, et. al., 1992]. Others prefer Denavit-Hartenberg notation [Ehman and Wu, 1987]. Most researchers in precision machine design and analysis employ homogeneous transformation matrices (HTMs) [Slocum, 1992]. Soons employed linearized variants of HTMs arguing that any higher order terms were negligible in practice [Soons, 1993]. Any of these mathematical notations are acceptable means of determining the error motions of a cutting tool with respect to a workpiece.

Sensitive directions are a means of determining the effect of machine accuracy on workpiece accuracy [Donaldson, 1980]. Most researchers characterize the accuracy of a machine tool by estimating errors in tool point location [CIRP, 1978]. However, the success of a machining operation is determined by part geometry, not tool position accuracy. The accuracy of the workpiece is generally only affected by cutting tool motion in the sensitive direction. The sensitive direction is the direction normal to the surface of the workpiece at the point that the tool is nominally in contact with the workpiece (Figure 9.4.1.1). Any direction normal to the sensitive direction is defined as a non-sensitive direction. If the error component in the non-sensitive direction is small compared to both the cutting tool and workpiece radii, then the corresponding workpiece error will be small. Therefore, in most cases, it is reasonable to neglect the error component in the non-sensitive directions. Donaldson [1980] developed methods to calculate sensitive directions as a function of the surface slope of the part for single point cutting tools.

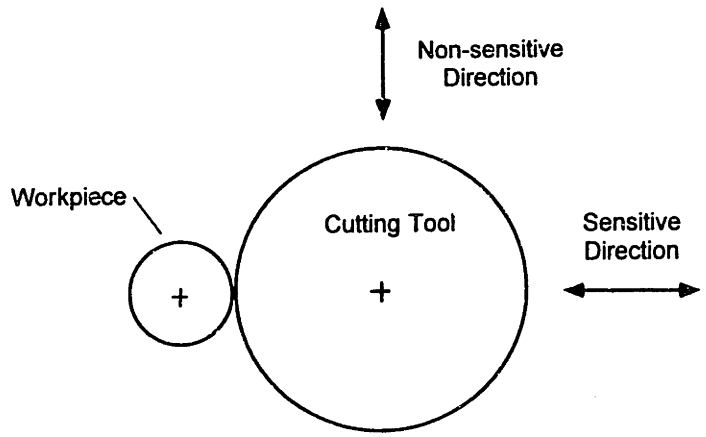


Figure 9.4.1.1 Sensitive directions in machining.

Error combination rules have been proposed to allow simultaneous consideration of the many noise factors that affect machine tool accuracy. If the noise factors are systematic (if they have the same values on repeated trials) then the effects may simply be added. However, it has been noted that addition of individual effects is often much too conservative. Many of the noise factors in a system can be modeled as random variables. It follows therefore, that the standard deviation of their effects combine according to a root sum of squares law rather than by addition. The need to estimate the combined effects of many noise factors has spawned research in combination rules for error. Slocum [1993] recommends that systematic errors be added, that random errors be combined by root sum of squares, and that the two should be combined later by averaging "worst case" and "best case" estimates. This is an empirical rule of thumb. It is not derived from probability theory, but has proven reasonable for practical use. Shen and Duffie [1993] compared this and other error combination schemes with the results of Monte Carlo simulation results for a measuring machine error budget. Park and Himmelblaum [1980] discussed the inadequacy of these rules when non-linear relationships exist between noise factors and experimental observables.

12.4.2 Bearing on this Thesis

Some aspects of this thesis can be considered an extension of current error budgeting. It extends current error budgeting techniques in the following ways:

- Error budgets for machines are usually formed with only a single criterion in mind. The methods developed in Chapter 5 allow one to consider multiple criteria simultaneously (e.g. size, profile, and surface finish tolerances).
- Error budgets are currently limited to analysis of a single machine tool. The system block diagrams presented in Chapter 6 allow one to estimate the performance of a system comprised of many machines.
- Current combination rules for error are not applicable to form tolerances. As described in Chapter 9, the concept of noise signatures and its combination with the Monte Carlo method allow one to quickly estimate the expected value of dimensions of form.
- The concept of sensitive directions is usually an adequate model for single point cutting operations. However, it is not applicable to cutting tools for which the geometric interaction of the tool and the work is more complex. The concept of sensitive directions may not be appropriate to apply if:
  1. The sensitive directions are difficult or impossible to define
  2. The location and motion of the cutting "point" are difficult or impossible to define
  3. The errors in the non-sensitive directions are large.

The first two criteria often arise in form grinding and milling. In these operations, there is no single sensitive direction. For example, Figure 12.4.2.1 depicts the profile of an engine poppet valve and the changes in sensitive direction changes along the profile. In plunge grinding a profile with a tool whose axis of symmetry is coplanar with the axis of symmetry of the workpiece, the curve of contact will lie within the plane of the tool and workpiece axes. In this case, it may be feasible to adapt the sensitive directions strategy by simply allowing the sensitive direction to vary along the workpiece profile. However, if the axis of symmetry of the cutting tool is **not** coplanar with the axis of symmetry of the workpiece, the curve of contact may not lie within a single plane. In this case, the computation of the curve of contact requires a model of the geometric interaction of the cutting tool with the workpiece and the sensitive directions are difficult to compute. This phenomenon was explored through the case study on valve grinding in chapter 10.

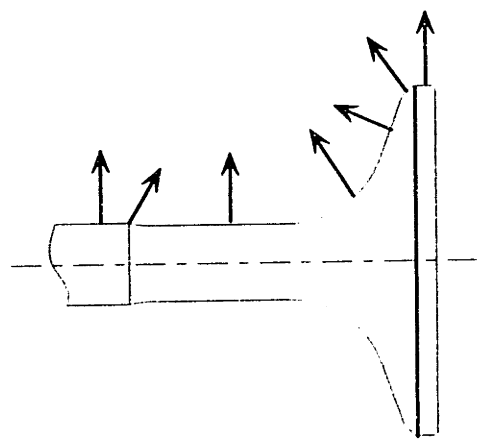


Figure 12.4.2.1 Changing sensitive directions along the profile of an engine valve.

The third case listed above may arise as a designer uses an error budget to seek more economical designs. Using the error budget as a guide, the designer is compelled to relax tolerances associated with error motions in the non-sensitive directions. In some cases the magnitude of the error motions in the non-sensitive directions can become so large that, despite the lower gains, they nevertheless contribute significantly to overall machine error. If the designer assumes *a priori* that non-sensitive directions are never important, the limits of this assumption cannot be discovered. This case is also illuminated by the example application in Chapter 10 in which the allowable attrition of a grinding wheel is relaxed causing a significant non-linearity in response a noise factor.

12.5 Axiomatic Design

12.5.1 Review of Axiomatic Design

Axiomatic Design (AD) is a theory of design proposed by Nam Suh (1978, 1991). Its purpose is to create a scientific basis for design. Its fundamental assumption is there are principles that every system that provides function must obey.

The axiomatic approach distinguishes Suh's theory from algorithmic approaches to the study of engineering design. An algorithmic approach defines step-by-step procedures for a designer to follow. Algorithmic approaches tend to be problem specific. For example, a knowledge based expert system for the design of VLSI circuits will not work well on the design of a an aircraft. By contrast, an axiomatic approach identifies primitive, true propositions and develops theorems from the primitive propositions by deduction. In taking an axiomatic approach, Suh intends to develop a science base for design of *anything* – manufacturing equipment, consumer products, software, even corporations.

The world of Axiomatic Design has four domains – the customer domain, the functional domain, the physical domain, and the process domain (see Fig 1). All design problems are viewed as mappings between these domains. In mapping between any two domains, the domain on the left is "what we want to achieve" and the domain on the right is "how we propose to achieve it." For example, the functional requirements (FRs) are the minimum set of independent requirements that characterize the design goals. The design parameters (DPs) are the key variables that characterize the physical entity created by the design process to fulfill the FRs. Therefore, the design of a product is the mapping from FRs to DPs. Similarly, the process variables (PVs) are the key variables that characterize the manufacturing process and determine the values of the DPs.

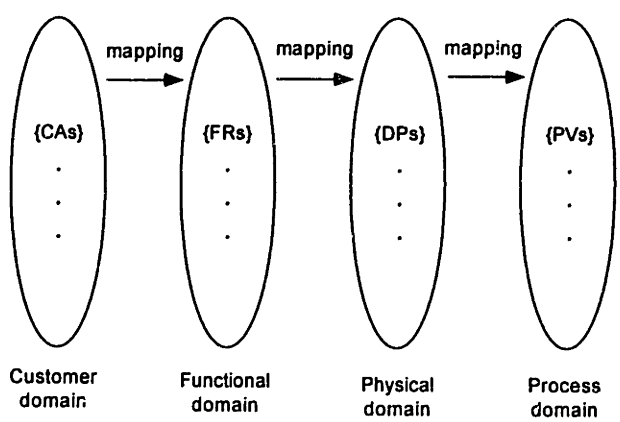


Figure 12.5.1.1 The four domains in the world of Axiomatic Design.

Given the framework defined above, Suh defines an axiomatic system. Suh defines axioms as "general principles or self-evident truths that that cannot be derived or proved to be true except that there are no counter-examples or exceptions." [Suh, 90] Theorems and corollaries logically derived from axioms will be sound when the above definition holds. Suh posits two axioms:

- The Independence Axiom – Maintain the independence of the functional requirements.
- The Information Axiom – Minimize the information content.

The Independence Axiom can be interpreted through the use of the design matrix, **A**. According to Suh, the mapping from the functional domain to the physical domain can be represented by the design equation

$$\{\mathbf{FR}\} = [\mathbf{A}]\{\mathbf{DP}\} \tag{12.5.1.1}$$

where

$$\mathbf{A}_{i,j} = \frac{\partial \mathbf{FR}_i}{\partial \mathbf{DP}_j}. \tag{12.5.1.2}$$

Suh defines an *uncoupled design* as a design whose **A** matrix can be arranged as a diagonal matrix by an appropriate ordering of the FRs and DPs. He defines a *decoupled design* as a design whose **A** matrix can be arranged as a triangular matrix by an appropriate ordering of the FRs and DPs. He defines a *coupled design* as a design whose **A** matrix **cannot** be arranged as a triangular or diagonal matrix by an appropriate ordering of the FRs and DPs. Suh states that an *uncoupled design* satisfies the Independence Axiom, that a *decoupled* design satisfies the Independence Axiom as long as changes in the DPs are performed in the appropriate order, and that a *coupled design* does not satisfy the Independence Axiom.

$$\underset{\text{uncoupled design}}{\begin{bmatrix} X & 0 & 0 \\ 0 & X & 0 \\ 0 & 0 & X \end{bmatrix}} \quad \underset{\text{decoupled design}}{\begin{bmatrix} X & X & X \\ 0 & X & X \\ 0 & 0 & X \end{bmatrix}} \quad \underset{\text{coupled design}}{\begin{bmatrix} X & X & X \\ X & X & X \\ X & X & X \end{bmatrix}}$$

Figure 12.5.1.2 Categories of design based on the structure of the design matrix.

For a matrix to be diagonal or triangular, some of the elements of that matrix must be identically zero. This requirement is relaxed by considering tolerance in the definition of coupling. The criterion for considering the non-diagonal elements to be close enough to zero is given by Theorem 8 below [Suh, 1990].

Theorem 8 (Independence and Tolerance) – A design is an uncoupled design when every designer specified tolerance $\Delta\mathbf{FR}_i$ is greater than

$$\Delta\mathbf{FR}_i > \sum_{\substack{j \neq i \\ j=1}}^{n} \left(\frac{\partial \mathbf{FR}_i}{\partial \mathbf{DP}_j} \right) \Delta\mathbf{DP}_j \qquad (12.5.1.3)$$

in which case the non-diagonal elements of the design matrix can be neglected from design consideration.

This concept is applied to manufacturability of a design by defining the mapping from the design domain to the process domain as

$$\{\mathbf{DP}\} = [\mathbf{B}]\{\mathbf{PV}\}. \qquad (12.5.1.4)$$

Given this representation of manufacture, Suh formulated the following theorem [Suh, 1990].

Theorem 9 (Design for Manufacturability) – For a product to be manufacturable, the design matrix of a product, [**A**] ... times the design matrix for the manufacturing process, [**B**] ... must yield either a diagonal or triangular matrix. Consequently, when ... either [**A**] or [**B**] represents a coupled design, the product cannot be manufactured.

The information axiom depends on Suh's definition of the information content of a design. The probability that a product can satisfy *all* of its FRs is called the probability of success (p_S). Based on the notion of probability of success, information content I is defined as

$$I = \log_2(1/p_s) \qquad (12.5.1.5)$$

Since the information measure defined above increases monotonically for all p_S, the Information Axiom can be restated as "select the design with the highest probability of meeting all of its FRs."

In the case that the probability density function over DP is uniformly distributed over the system range, and given that the tolerance on the functional requirements determines a tolerance range on the DP, then the information content can be expressed as

$$I = \log_2\left(\frac{\text{system range}}{\text{common range}}\right) \tag{12.5.1.6}$$

where the common range is the intersection of the system range and the design range (Figure 3).

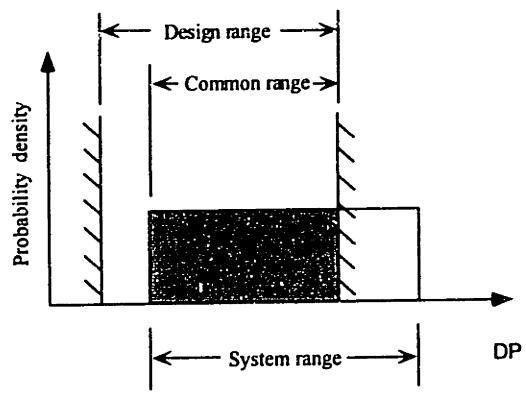


Figure 12.5.1.3 Information in the case of uniformly distributed variation.

If a set of events are statistically independent, then the probability of the union of the events is the product of the probabilities of the individual events. Given the logarithmic definition information content, a two theorems follows concerning the sum of information.

Theorem 12 (Sum of Information) – The sum of information for a set of events is also information, provided that the proper conditional probabilities are used when the events are not statistically independent.

Theorem 13 (Information Content of the Total System) – If each FR is probabilistically independent of other FRs, the information content of the total system is the sum of information of all individual events associated with the set of FRs that must be satisfied.

(NOTE: pg. 394 of *TPD* reads "If each DP is probabilistically independent of other DPs..." but page 154 gives the statement of the theorem above which I assume is the one intended.)

The independence axiom and the information axiom are interrelated. For example, if there exists a difference between the current FR values and the desired FR values, then the DPs may be used to adjust the FR values back to their targets. If the design is uncoupled, then each FR may be adjusted with its corresponding DP. The diagonal structure of the design matrix ensures that the adjustment of one DP will not interfere with the FR values that are already on target. This observation is captured in the following theorem.

Theorem 6 (Path Independency of Uncoupled Design) – The information contents of an uncoupled design is independent of the sequence by which the DPs are changed to satisfy the given set of FRs.

Decoupled designs may also be adjusted onto target by sequential adjustment of the DPs, however, the DPs must be adjusted in the appropriate order to ensure that the previously set FRs are not upset by later adjustments. Coupled designs cannot be adjusted by a sequential adjustment process. These observations are expressed the theorem below.

Theorem 7 (Path Dependency of Coupled and Decoupled Design) – The information contents of coupled and decoupled designs depend on the sequence by which the DPs are changed and on the specific paths of change of these DPs.

The inter-relationship of the Independence and the Information Axioms is perhaps most strongly stated in the following theorem.

Theorem 14 (Information Content of Coupled versus Uncoupled Designs – When the state of FRs is changed from one state to another in the functional domain, the information required for the change is greater for a coupled process than for an uncoupled process.

This section outlined the basics of Axiomatic Design. The following section will examine the relationship of Axiomatic Design to the theory presented in this dissertation.

12.5.2 Bearing on this Thesis

This thesis has been strongly influenced by Axiomatic Design. The basic approach to representing a manufacturing system as a matrix is one obvious similarity. There are however some important differences in the foundations of the two theories. Ultimately, these differences lead to theorems in this thesis distinct from those in Axiomatic Design.

Axiomatic Design and this thesis have very different foundations. Suh proposed *new axioms* by generalizing from empirical evidence. The rest of Axiomatic Design is based on the axioms by deduction. Thus, Axiomatic design is claimed to be a new science which relies, as do all other sciences, on the validity of inductive inferences and experimental data and is subject to falsification by counter-example. By contrast, this thesis proposes no new axioms. All of the theorems proven here are based upon the axioms of probability, classical real analysis, and Euclidean geometry.

Axiomatic Design and this thesis also differ in scope. Axiomatic Design has been applied to a wide range of phenomena in engineering design, management, software engineering, and many other fields. This thesis by contrast is strictly limited in its range of application by a set of explicit assumptions. This thesis is intended for use in analyzing variation in manufacture and makes no claims outside this area.

Axiomatic Design and this thesis also differ in subject matter (even when they apply in the same field). Axiomatic design and this thesis both posit theorems applicable to variation and probability in manufacture. Though the two theories both apply to manufacture, they differ substantially in the phenomena to which they refer as one can see by examining the definitions they employ. For example, Suh's "design parameters" (DPs) are different from the "quality characteristics" defined in this thesis. Suh defines DPs as the key variables that characterize the physical entity created by the design process to fulfill the FRs. In this thesis, a "quality characteristic" of a discrete unit of product are the complete set of dimensions whose tolerances define the product's acceptance criteria.. Given my definition of quality characteristics, they must include (for example) the cylindricity, size, and surface finish of an automotive crankpin if

such tolerances are specified on the engineering drawing of the crankshaft. The documentation of a crankshaft will generally not indicate how (or whether) each of these dimensions fulfills separate functions of the crankshaft. So, the cylindricity, size, and surface finish of an automotive crankpin may or may not be considered examples of Suh's DPs.

Another important difference lies in the inputs to the matrix. There are important distinctions between a "process variable" as defined by Suh and my definition of a "noise factor." In Suh's framework, process variables (PVs) are those variables chosen by the designer to determine the DPs. In this thesis, a noise factor of a manufacturing system is a quantity whose variation causes variation in the quality vector. Therefore, under the definition used here, the ambient temperature within a manufacturing facility might be included as a noise factor if its variation affects the dimensional accuracy of the product. In Axiomatic Design, ambient temperature would not normally be listed as a PV.

Given the broad definition of quality characteristics and noise factors, many DPs are quality characteristics and many PV are noise factors. Therefore, many theorems proven in this dissertation will apply to DPs and PVs in Axiomatic Design. However, there exist quality characteristics which are not DPs and noise factors which are not PVs. Therefore, some of the theorems of Axiomatic Design may not apply to quality characteristics and noise factors as defined here. These two points will are discussed further below.

Because of the overlap between DPs and PVs and quality characteristics and noise factors, much of this thesis can be viewed as an extension of Axiomatic Design. For example, Theorem 13 of AD (Information Content of the Total System) applies *only to probabilistically independent* FRs. It follows from probability calculus (as shown in Section 5.6) that DPs will not be probabilistically independent if **B** is diagonal and the PVs are probabilistically independent. For the same reasons, FRs will not be probabilistically independent for decoupled **A** matrices with probabilistically independent DPs. Therefore, information content cannot be summed for decoupled designs. None of the literature on Axiomatic Design address the computation of the information content of decoupled designs. Chapter 5 of this dissertation provide algorithms that can be used to compute the probability of success for decoupled designs given a variety of different forms of probability density functions over PV. This is just one example of how this thesis can be employed within the framework of Axiomatic Design.

An obvious similarity between Axiomatic Design and this thesis is in their representation of manufacture. The process capability matrix **C** bears a strong resemblance to the **B** matrix defined by Suh. Both matrices represent a transformation from variables describing a manufacturing system to variables describing a product. There are some important differences however. One is that the variables being transformed are different (as discussed above). Another possible difference is that, in this thesis, the behavior of the system is *assumed* to be linear over the range of interest. Therefore, the linear transformation **C** is quite literal. This restricts the domain of applicability of this thesis. Axiomatic Design is not similarly restricted; Suh considers non-linear behavior to fall within the purview of axiomatic design.

In taking the linear transformation **C** as literal, this thesis can employ all of the mathematical devices of linear algebra. Thus, the generalized inverse of the **C** matrix is meaningful as in inverse mapping (or a least squares mapping) from dimensions of the product to process variables. This means that the restriction to sequential adjustment of process variables implied by Theorem 7 of Axiomatic Design does not apply to the adjustment variables discussed in Chapter 7. Subject to the assumptions of this thesis, a settings of the adjustment variables may be determined "all at once" (as opposed to sequentially) based on known variations in the quality characteristics using the generalized inverse of **C**.

Finally, I must emphasize that care should be used when applying theorems of Axiomatic Design to quality characteristics and noise factors as defined in this thesis. For example, Theorem 9 of Axiomatic Design might be misinterpreted as meaning that non-triangular **C** matrices have an upper limit to their rolled throughput yield, Y_{RT}. However, it possible to construct fully populated **C** matrices that have Y_{RT} that approach unity. It is important to be cognizant of the differences in the definition of the term "process variable" that explain this apparent contradiction. Much of this thesis may appear surprising when viewed from an Axiomatic Design perspective.

This thesis differs from Axiomatic Design in its foundations, scope, and subject matter. Still, there is considerable overlap; It is possible that the theorems, equations, and algorithms presented here will be useful to practitioners of Axiomatic Design for computing the probability of success of decoupled designs.

12.6 Tolerance Analysis and Synthesis

12.6.1 Review of Tolerance Analysis and Synthesis

Assignment of tolerances to components within an assembly is a key task in design for manufacture. Tolerances must be loose enough to ensure manufacturability of the parts and tight enough to guarantee proper function of the assembly. Proper tolerance assignment demands good tolerance analysis and synthesis.

Tolerance analysis is primarily concerned with estimation of variance of a dimension of an assembly given estimates of variation in the components of an assembly. The basics of this well developed field were summarized by Evans [1974]. Some valuable contributions to the theory have been made more recently [Bjorke, 1978, Greenwood and Chase, 1987, Whitney, et. al., 1994]. There also exists very effective commercial software for solving tolerance stack up problems. Among the most widely used packages is VSA-3D (developed by Variation Systems Analysis, Inc., St. Clair Shores, MI) which employs Monte Carlo simulation to estimate yield, variance, and sensitivity to design parameters [Craig, 1991]. Other statistical simulation packages have been developed by universities [Lehtihet and Dindelli, 1989, Turner, 1991]. Straight Monte Carlo simulation is considered by some to be inadequate for production problems due to the large number of trials required to achieve accurate results and because of the high noise in tail behavior [Keeler, et al, 1994].

Especially relevant to this thesis is analysis of assemblies with interrelated tolerances chains. Bjorke [1978] classifies interrelated tolerance chains into two groups: (1) Loops with common links and (2) Loops with common probability. Loops with common links are interrelated by the fact that they share links and therefore, the dimensions of the loops are correlated. Loops with common probability are interrelated by the necessity for simultaneous satisfaction of tolerances on the loop dimensions. Bjorke provides closed form techniques for dealing with both of these cases. He does not, however, provide means to deal with tolerance loops that are related by *both* common links *and* common probability. These two categories are not mutually exclusive.

Tolerance synthesis combines tolerance analysis with optimization theory to compute a set of tolerances that are in some sense optimal. Ostwald and Huang [1977] applied linear programming to the problem of least cost tolerance allocation by assuming a linear cost model

and taking tolerances as linear inequality constraints. Michael and Siddall [1981] formulated tolerance allocation problems with multiple constraints but assuming that full acceptance is required. Lee and Woo [1989] employed a branch and bound algorithm to select the optimum from a space of discrete tolerances. Dong and Soom [1990] created an expert system for automatic tolerance allocation that includes a gradient based optimization algorithm and an IGES interface.

12.6.2 Bearing on this Thesis

Although there are no assembly case studies in this thesis, the equations and algorithms developed here may be applied to a wide class of tolerance analysis and synthesis problems.

The kinematic modeling methods developed in Chapter 9 may be used to form models of assemblies not unlike the vector loops employed by Chase and Greenwood [1987]. Such a kinematic model can be used to determine the sensitivity of key dimensions of an assembly to noise factors such as misalignments and component dimensions.

The sensitivities derived from the kinematic models may be used to form a process capability matrix as described in Chapter 3. If capability matrices are formed for subassembly operations, then a model of the total assembly operation can be formed using block diagramming procedures as described in Chapter 4. The capability matrix for can then be used to compute assembly yields as described in Chapter 5. The linearity assumption in the capability matrix approach restricts consideration to assemblies in which misalignments are below about five degrees. If there is only one important dimension on the assembly, then the methods of this paper begin to resemble a simple root sum of squares stack up analysis combined with a use of z-values to compute yield. However, if there are multiple, correlated toleranced dimensions, then the equations and algorithms of Chapter 5 may provide a substantial improvement over existing methods. This approach is especially useful for interrelated dimensional chains with *both* common links *and* common probabilities (as defined by Bjorke [1978]).

Some of the tolerancing literature raises important questions about computation of probabilities. Stephen Keeler at Boeing has correctly identified the weaknesses of Monte Carlo simulation and its use in statistical pre-assembly for tolerance analysis [Keeler, et al., 1994]. This thesis addresses this concern to some extent by developing closed-form solutions for a few

important special cases. However, for high-dimensional problems, some form of probabilistic computation method (such as the Monte Carlo method) is often a very valuable means of computing approximate results.

One of the most important applications of this thesis in assembly may be the adjustment modeling techniques in Chapter 7. Many assembly operations include adjustment operations such as shimming and alignment operations. I propose that the block diagramming procedures can capture the effect of these on-line adjustments and the importance of the ordering of assembly steps. Further case studies will be required to validate this conjecture for the application to assembly procedures.

Another possible application of this work in assembly modeling is expansion of assembly models to include process information. Assembly modeling programs such as VSA require input of probabilistic information on component accuracy. The methods of this thesis should allow one to expand the model one step further. One may include a process capability matrix for the component manufacturing process into the assembly model. In effect, the capability matrix will generate the appropriate joint density function for the component dimensions based on probabilistic models of the noise factors affecting the component manufacturing process. The results of this thesis can therefore be considered an extension of statistical pre-assembly approaches (such as VSA) to include the manufacture of the components of an assembly.

13. Conclusions

This thesis provides a mathematical analysis of the sources, flow, and effects of variation in production systems. I demonstrate that manufacturing systems can be viewed as multi-input multi-output systems with dominance and correlation. Manufacturing systems may include multiple processing steps, on-line adjustments, and inspection procedures. The multiple inputs are noise factors in the environment, equipment, and materiel. The multiple outputs are quality characteristics of products. Dominance is the property that a failure to meet a single tolerance on a discrete unit of product results in scrap or rework. Correlation refers to the information gained about one quality characteristic by measuring another quality characteristic.

I proposed a new way to represent manufacturing systems especially suited to designing systems that meet product acceptance criteria with high probability. Process capability matrices and bias vectors represent the response of processing steps to noise factors. I show that these two structures capture information on noise signal amplitude, system sensitivity, tolerance of variation, correlation, and mean shifts and drifts. The capability matrix approach has the significant limitation that it assumes approximately linear behavior within the range of likely noise factor values. These matrix / vector pairs are useful in block diagram representations of manufacturing systems that pictorially represent the flow of variation through complex manufacturing systems including multiple processing steps and on-line adjustments. Reduction rules permit complex systems to be represented by a single capability matrix and bias vector. I developed several algorithms and equations that map a system's capability matrix and bias vector into an estimate of rolled throughput yield.

An industrial case study concerning surface mount of multichip modules demonstrated the value of the capability matrix as a tool for design decision making. The surface mount case study shows that correlation can have a first order effect on system yield. Industry standard producibilty analysis techniques, such as six sigma methodology, fail to account for correlation and therefore predict yields in error by over two orders of magnitude. The capability matrix approach accurately accounts for the influence of correlation and can efficiently compute the first few significant digits of rolled throughput yield. The surface mount case study also clearly

demonstrates the power of the capability matrix approach in modeling on-line adjustment strategies and their ability to reduce variance and bias.

This thesis provides a detailed framework for modeling machining processes (I call it "virtual machining"). The techniques allow one to model the effects of noise factors on machining accuracy as a cutting tool sweeps through its workspace. Homogeneous transformation matrix (HTM) models represent the effects of machine error motions on the position of carriages and other machine elements. An original swept envelope algorithm efficiently computes points on the surface of the workpiece based on the cutting tool's motion function. The framework may be used to create databases of error signatures that capture the pattern of error across a workpiece surface due to various noise factors. Monte Carlo simulation is used to superpose error signatures for the purpose of error budgeting.

A case study of dual head grinding of engine poppet valves serves as an introduction to the virtual machining framework The kinematic modeling methods automatically account for Abbé error. The swept envelope algorithm a shown to capture the non-linear effects of extended material removal processes.

A case study of CNC crankpin grinding serves as a more comprehensive case study on both virtual machining and process capability matrices. A computerized virtual machining model is used to develop a database of error signatures. These error signatures were employed in fault diagnosis to bring about a threefold improvement in machine accuracy one case. The error signatures reveal the non-linear behavior of certain tolerances (e.g., roundness). The process capability matrix approach is shown to compare favorably with Monte Carlo error budgeting despite these non-linearities.

References

Anderson, David M., 1990, *Design for Manufacturability*, CIM Press, Lafayette, CA.

ASME, 1983, *ANSI Y14.5M -- Dimensioning and Tolerancing*, Americain Society of Mechanical Engineering, New York.

Ben-Israel, Adi, and Thomas N. E. Greville, 1980, *Generalized Inverses: Theory and Applications*, Robert E. Krieger Publishing Company, Huntington, NY.

Bhote, K., R., 1991, *World Class Quality*, AMACOM, New York.

Bjorke, O., 1978, *Computer Aided Tolerancing*, Tapir Publishers, Trondheim, Norway.

Box, G. E. P., and K. B. Wilson, 1951, "On the Experimental Attainment of Optimum Conditions," *Journal of the Royal Statistical Society*, B 13, 1.

Bryan, J. B., 1984, "The Power of Deterministic Thinking in Machine Tool Accuracy," *First International Machine Tool Engineers Conference*, Tokyo.

Bryan, J. B., 1989, "The Abbé Principle Revisited – An Updated Interpretation," *Precision Engineering*, vol. 1, no. 3, pp.129-132.

Buslenko, N. P., D. I Golenko, Yu. A. Scheider, I. M. Sobol, and V.G. Sragovich, 1966, *The Monte Carlo Method: The Method of Statistical Trials*, Pergamon Press, Oxford.

Chao, L. M. and J. C. S. Yang, 1987, "Implementation of a scheme to improve the positioning accuracy of an articulate robot by using laser distance-measuring interferometry," *Precision Engineering*, vol. 9, no. 4, pp. 210 - 217.

CIRP, 1978, "A Proposal for Defining and Specifying the Dimensional Uncertainty of Multiaxis Measuring Machines," *Annals of the CIRP*, Vol. 27, No. 2, pp 623-630.

CIRP, 1979, "Three Dimensional Uncertainty," *Annals of the CIRP*, vol. 28, no. 2.

Clifford, A. A., 1973, *Multivariate Error Analysis: A Handbook of Error Propagation and Calculation in Many-Parameter Systems*, John Wiley & Sons, New York.

Cook, Robert D, David S. Malkus, and Micheal E. Plesha, 1974, *Concepts and Applications in Finite Element Analysis*, John Wiley & Sons, New York.

Cover, Thomas M., and Joy A. Thomas, 1991, *Elements of Information Theory*, John Wiley & Sons, New York.

Craig, M., 1988, "Variation by Design," *Mechanical Engineering*, vol. 110, no. 11, pp. 52-54.

Danielson, Hans, 1995, *Surface Mount Technology with Fine Pitch Components: The manufacturing Issues*, Chapman & Hall, London.

Denavit, J. and R. Hartenberg, 1955, "A Kinematic Notation for Lower Pair Mechanisms Based on Matrices", *Journal of Applied Mechechanics*.

Doane, Daryl Ann, and Paul D. Franzon, 1993, *Multichip Module Technologies and Alternatives: The Basics*, Van Nostrand Reinhold, NewYork.

Donaldson, R. R., 1980, "Error Budgets", *Technology of Machine Tools*, vol. 5, Machine Tool Task Force, Robert J. Hocken, Chairman.

Donaldson, R. R., and D. C. Thompson, 1986, "Design and Performance of a Small Precision CNC Turning Machine", *Annals of the CIRP*, vol. 35, no. 1, pp. 373-376.

Dong, Zuomin, and Andres Soom, 1990, "Automatic Optimal Tolerance Design for Related Dimensional Chains," *Manufacturing Review*, vol. 3, no. 4, pp. 262-271.

Donmez, Alkan, *A General Methodology for Machine Tool Accuracy Enhancement: Theory, Application, and Implementation*, PhD thesis, Purdue University, 1985.

Drake, Alvin W., 1967, *Fundamentals of Applied Probability Theory*, McGraw-Hill, New York.

Draper, N. R., and H. Smith, 1966, *Applied Regression Analysis*, John Wiley & Sons, New York.

Ehman, K. F., and B. T. Wu, 1987, "A Generalized Geometric Error Model for Multi-Axis Machines," *Annals of the CIRP*, vol. 36, no. 1.

European Cooperation for Accreditation of Laboratories, 1995, *Coordinate Measuring Machine Calibration*, EAL-G17.

Evans, D. H., 1974, "Statistical Tolerancing: The State of the Art, Part 2. Methods of Estimating Moments," *Journal of Quality Technology*, vol. 7, no. 1, pp.1-12.

Fisher, R. A., 1935, *The Design of Experiments*, Oliver and Boyd, London.

Frey, D. D., K. N. Otto, and W. Pflager, 1997a, "Swept Envelopes of Cutting Tools in Integrated Machine and Workpiece Error Budgeting", to appear in *Annals of the CIRP*.

Frey, D. D., K. N. Otto, and J. Wysocki, 1997b, "Evaluating Process Capability Given Multiple Acceptance Criteria", submitted to *ASME Journal of Manufacturing Science and Engineering*.

Frey, D. D., and T. Hykes, 1997c, *A Method for Virtual Machining*, U. S. Patent Application #08-804-545, granted, to issue.

Fridshal, R., K. P. Cheng, D. Duncan, and Z. Zucker, "Geometric Modeling Based Numerical Control Part Program Verification," *Proc. Numerical Control Society, 19th Ann. Meeting and Tech. Conf.*, Apr. 1982.

Goldratt, Iliahu, 1989, *The Goal*, North River Press, New York.

Greenwood, W. H. and K. W. Chase, 1987, "A New Tolerance Analysis Method for Designers and Manufacturers," *ASME Journal of Engineering for Industry*, vol. 109, pp. 112-116.

Guo, Ruey-Shan, and Emanuel Sachs, 1993, "Modeling, Optimization, and Control of Spatial Uniformity in Manufacturing Processes," *IEEE Transactions on Semiconductor Manufacturing*, vol. 6, no. 1.

Harry, Mikel J., and J. Ronald Lawson, 1992, *Six Sigma Producibility Analysis and Process Characterization*, Addison Wesley, Reading, MA.

Hughes Electronics Company, 1995, *6σ Design for Manufacturability: An Overview*, EOS Six Sigma Training Presentation.

Hughes Electronics Company, 1994, *Six Sigma / DFMA*, license agreement GMH L-9451.

Ishikawa, Kaoru, 1982, *Guide to Quality Control*, Asian Productivity Organization, Tokyo.

Jaikumar, Ramachandran, 1988, *From Filing and Fitting to Flexible Manufacture*, working paper 88-045.

Juran, J. M., 1951, *Juran's Quality Control Handbook*, McGraw Hill, New York.

Kahino, Y. , Y. Ihara, and A. Shinohara, 1993, *Accuracy Inspection of NC Machine Tools by Double Ball Bar Method*, Hanser Publishers, Munich.

Kalpakjian, Serope, 1995, *Manufacturing Engineering and Technology*, Addison Wesley, NY.

Kane, V. E., 1986, "Process Capability Indices," *Journal of Quality Technology*, Vol. 18, No. 1, pp. 41-52.

Kazmer, D., Barkan, P., and K. Ishii, 1996, "Quantifying Design and Manufacturing Robustness Through Stochastic Optimization Techniques," *ASME Design Technical Conferences*, Irvine, CA.

Keeler, Stephen P., Alan K. Jones, and Harold A. Scott, 1994, "Tolerancing Methods and Software: A Status Report," Document No. BCSTECH-94-030, Boeing Computer Services, Seattle, WA.

Knapp, W., and S. Hrovat, 1987, *The Circular Test*, S. Hrovat (Publisher), Zurich, Switzerland.

Knapp, W., 1983, "Circular Test for Three-Coordinate Measuring Machines and Machine Tools," *Precision Engineering*, vol. 5, no. 3, pp. 115-124.

Lee, W. J., and T. C. Woo, 1989, "Optimum Selection of Discrete Tolerances," *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 111, pp. 243-251.

Lehtihet, E. A., and B. A. Dindelli, 1989, "TOLCON: Microcomputer Based Module for Simulation of Tolerances," *Manufacturing Review*, vol. 2, no. 3.

Lin, P. D., and K. F. Ehmann, 1993, "Direct Volumetric Error Evaluation for Multi-Axis Machines," *International Journal of Machine Tools and Manufacture*, vol. 33, no. 5, pp. 675-693.

Martin, R. R., and P. C. Stephenson, 1990, "Sweeping of Three-Dimensional Objects," *Computer Aided Design*, vol. 22, no. 4, pp. 223-234.

Meyers, Raymond H.,1971, *Response Surface Methodology*, Allyn and Bacon, Boston.

Michael, W., and J. N. Siddall, 1981, "The Optimization Problem with Optimal Tolerance Assignment and Full Acceptance," *ASME Journal of Mechanical Design*, vol. 103, pp. 842-848.

Monge, G., 1851, "Application de L'Analyse a le Geometrie, Section VII," *Des surfaces des canaux dont l'axe est une courbe queleconque plane et horizontale, et dont les sections perpendiculaires a cet axe sont des cercles de rayon constant*, ed. M. Liouville, Bachelier publishers, Paris, France.

Murthy, T. S. R., 1986, "A Comparison of Different Algorithms for Circularity Evaluation," *Precision Engineering*, vol. 8, no. 1, pp. 19-23.

Narawa, L., M. Kowalski, and J. Sladek, 1989, "The Influence of Kinematic Errors on the Profile Shapes by Means of CMM," *Annals of the CIRP*, vol. 38, no. 1.

Ogata, K., 1970, *Modern Control Engineering*, Prentice-Hall, Englewood Cliffs, New Jersey.

Ostwald, P. F., and J. Huang, 1977, "A Method for Optimal Tolerance Selection," *ASME Journal of Engineering for Industry*, vol. 99, pp. 558-565.

Otto, K., 1994, "Robust Product Design Using Manufacturing Adjustments," *ASME Design Theory and Methodology Conference*, Minneapolis, MN.

Otto, K. and E. Antonsson, 1993, "Tuning Parameters in Engineering Design," *Journal of Mechanical Design*, vol. 115, pp. 14-19.

Park, S. W., and D. M. Himmelblaum, 1980, "Error in the Error Propagation Formula," *AIChE Journal*, vol. 26, no. 1, pp. 168-170.

Penrose, R., 1955, "A Generalized Inverse for Matrices," *Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406-413.

Phadke, Madhav S., 1989, *Quality Engineering Using Robust Design*, Prentice Hall, Englewood Cliffs, New Jersey.

Portman, T., 1980, "Error Summation in the Analytical Calculation of Lathe Accuracy," *Machines and Tooling*, vol. 51, no. 1, pp. 7-10.

Pugh, Stuart, 1991, *Total Design: Integrated Methods for Successful Product Engineering*, Addison-Wesley, Wokingham, England.

Renishaw Corporation, 1995, *QC10 Ballbar diagnostic software.*

Shen, Y. L., and N. A. Duffie, 1993, "Comparison of Combinatorial Rules for Machine Error Budgets," *Annals of the CIRP*, vol. 42, no. 1, pp.619-622.

Shewhart, W. A., 1931, *Economic Control of Quality of Manufactured Products*, D. Van Nostrand, New York.

Shiba, Shoji, Alan Graham, and David Walden, 1993, *A New American TQM: Four Practical Revolutions in Management*, Productivity Press, Portland, OR.

Slocum, A. H., 1992, *Precision Machine Design*, Prentice-Hall, Englewood Cliffs, New Jersey.

Soons, J. A., F. C. Theuws, and P.H. Schellekens, 1992, "Modeling the Errors of Multi-Axis Machines: A General Methodology", *Precision Engineering*, vol. 14, no. 1, pp. 5-19.

Soons, J. A., 1993, *Accuracy Analysis of Multi-Axis Machines*, Ph.D. Thesis, Eindhoven University of Technology, The Netherlands.

Spiewak, S. A., 1987, "Analytical Modeling of Cutting Point Trajectories in Milling," *ASME Journal of Engineering for Industry*, Vol. 116, pp. 440-447.

Strauss, Rudolf, 1994, *Surface Mount Technology*, Butterworth Heinemann, Oxford.

Suh, N. P., 1990, *The Principles of Design*, Oxford University Press, Oxford.

Taguchi, G., 1987, *System of Experimental Design*, Vols. 1 and 2, ASI & Quality Resources, White Plains, New York.

Treib, T., 1987, "Error Budgeting - Applied to the Calculation and Optimization of the Volumetric Error Field of Multiaxis Systems", *Annals of the CIRP*, vol. 36, no.1, pp.365-368.

Turner, J. U., 1991, "Research in Automated Tolerancing at RPI," *Proceedings of the NSF Grantees Conference*, Austin, TX.

Vickers, G.W., "Ball-Mills versus End-Mills for Curved Surface Machining," *ASME Journal of Engineering for Industry*, Vol. 111, pp. 22-26.

Whitney, D. E., O. L. Gilbert, and M., Jastrzebski, 1994, "Representation of Geometric Variations Using Matrix Transforms for Statistical Tolerance Analysis in Assemblies," *Research in Engineering Design*, vol. 6, pp. 191-210.

Wang, W. P., and K. K. Wang, 1986, "Geometric Modeling for Swept Volume of Moving Solids," *IEEE Computer Graphics and Applications*, vol. 6, no. 12, pp. 8-17.

Ziegert, J. C. , D. G. Olson, and P. Datseris, 1992, "Description of Machine Tool Errors using Screw Coordinates," *Mechanical Design*, vol. 114, pp.531-535.

A. Glossary

Set Theory

- Set – A collection of distinguishable objects.
- Element – An object, a, which belongs to set, A, is said to be an element of the set.

$$a \in A$$

- Subset – If every element of A is also an element of B then A is a subset of B.

$$A \subseteq B$$

- Union – The set of all elements of A or B or both.

$$A \cup B$$

- Intersection – The set of all elements common to both A and B.

$$A \cap B$$

- Empty set – The set containing no elements. It is denoted by the symbol $\varnothing$.
- Universal set – The set of all objects under consideration in any given situation. It is denoted by the symbol S.
- Partition – A set of disjoint and exhaustive subsets of a given set.

Mathematics

- Function (or Transformation or Mapping) – A relation between two sets that associates a unique element of the second with each element of the first. Formally, a set of ordered pairs $\langle x, f(x) \rangle$. If we write $y = f(x)$, then y is the value of the function for the argument x. If the arguments x are elements of S and the values y are elements of A, then we write $f: S \to A$ or $f: x \mapsto y$.
- Domain (Maximal Domain) – The set S over which a function is defined.
- Image – The image of a set A is the range of values of a function taken for the elements in that set, written $f(A)$;

$$f(A) = \{y : y = f(x),\ x \in A\}$$

- Range – The set of values that a function takes over its domain. Formally, the image of the domain or $f(S)$.

- Graph – A drawing showing the functional relationship among variables. The set of points $\langle x, y \rangle$ where $y = f(x)$.
- Convolution – A function derived from two given functions f and g by the integration

$$\int_0^x f(t)g(x-t)\mathrm{d}t$$

Probability

- Experiment – A set of measurements taken under prescribed conditions.
- Trial – A single performance of the experiment.
- Outcome – The resulting measurements from a single trial.
- Sample space – The set of all possible outcomes of a given experiment. It is denoted by the symbol S. The universal set has the same symbol and is closely related.
- Event – A subset of the sample space, denoted here as A.
- Probability – A measure or estimate of the degree of confidence in the occurrence of an event. More precisely, a measure P defined over the sample space which obeys the axioms of probability.
- Axioms of Probability –

  axiom 1: $P(A) \geq 0$ The probability of any event is non-negative.

  axiom 2: $P(S) = 1$ The probability of the certain event is unity.

  axiom 3: $P\left(\bigcup_{n=1}^{N} A_n\right) = \sum_{n=1}^{N} P(A_n)$ if $A_m \cap A_n = \varnothing$ for all $m \neq n = 1,2,\ldots N$

  The probability of the union of mutually exclusive events is the sum of the individual event probabilities.
- Statistical Independence – Events A and B are said to be statistically independent if and only if

$$P(A \cap B) = P(A)P(B)$$

- Conditional Probability – The conditional probability of A given B, $P(A|B)$, is the probability of the event A subject to the hypothesis of the occurrence of event B and is defined formally as

$$P(A|B) = \frac{P(A \cap B)}{P(B)}.$$

- Bayes Theorem – A result useful in computing an updated probability $P(A_n|B)$ of a prior estimate of probability $P(A_n)$ of an event A_n in light of observation of an event B.

$$P(A_n|B) = \frac{P(B|A_n)P(A_n)}{\sum_{i=1}^{m}\left[P(B|A_i)P(A_i)\right]}$$

  where the set A_i constitutes a partition of the sample space.

- Random variable – A quantity that may take any of a range of values in multiple trials of an experiment. More formally, a real random variable x is a mapping from the sample space S to the real line $\Re$.
- Statistical independence of continuous random variables – Random variables x and y are said to be statistically independent if and only if

$$f_{xy}(x,y) = f_x(x)f_y(y)$$

- Random vector – An n dimensional real vector $\mathbf{x}$ is mapping from the sample space S to $\Re^n$.
- Probability density function – A function $f_X(x)$ representing the relative distribution of the frequency of a continuous random variable x and having the properties

$$0 \le f_x(x) \text{ for all } x$$

$$\int_{-\infty}^{\infty} f_x(x)\mathrm{d}x = 1$$

$$P\{a < x \le b\} = \int_a^b f_x(x)\mathrm{d}x$$

  Its graph is the limiting case of a histogram as the number of trials increases.

- Joint Density Function -- A function $f_{xy}(x,y)$ representing the relative distribution of the frequency of continuous random variables x and y and having the properties

$$0 \le f_{xy}(x,y) \text{ for all } x, y$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{xy}(x,y)\mathrm{d}x\mathrm{d}y = 1$$

$$P\{a < x \le b,\ c < y \le d\} = \int_c^d \int_a^b f_{xy}(x,y)\mathrm{d}x\mathrm{d}y$$

$$f_y(y) = \int_{-\infty}^{\infty} f_{xy}(x,y)\mathrm{d}x$$

The concept of joint density can be generalized to higher dimensions.

- Expected value – A measure of the central tendency of a function $g(x)$ of a random variable x.

$$E(g(x)) = \int_a^b g(x) f_x(x)\mathrm{d}x$$

- Mean – A measure of central tendency μ of a random variable x. Similar to an arithmetic mean but defined for a continuous random variable x as

$$\mu = E(x).$$

- Moment – The expected value of the power of the deviation of a random variable from its mean. The order of the moment is the power so that the nth order moment is

$$E((x - E(x))^n).$$

- Standard Deviation -- A measure σ of the dispersion of a random variable x defined as the square root of the second moment.

$$\sigma = \sqrt{E((x - E(x))^2)}$$

- Normal Distribution – A continuous probability distribution for which all moments of order higher than two are zero. The graph of the pdf of a normally distributed random variable x is

$$f_x(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Covariance Matrix -- The covariance matrix $\mathbf{K}$ defines correlations among random variables $\mathbf{x}_i$.

$$\mathbf{K}_{ij} = E((\mathbf{x}_i - E(\mathbf{x}_i))(\mathbf{x}_j - E(\mathbf{x}_j)))$$

- Joint Normal Distribution – A joint probability density function over random variables $\mathbf{x}_i$ is said to be normal if it can be written as

$$f_{x_1,\ldots x_n}(\mathbf{x}_1,\ldots\mathbf{x}_n) = \frac{\sqrt{|\mathbf{K}^{-1}|}}{(2\pi)^{n/2}}\exp\left\{-\frac{(\mathbf{x}-E(\mathbf{x}))^T\mathbf{K}^{-1}(\mathbf{x}-E(\mathbf{x}))}{2}\right\}$$

- Central Limit Theorem – The mean of a sequence of n independent identically distributed random variables of finite mean μ and $E\left[|x_i - E(x_i)|^{2+\delta}\right] < \infty \quad \delta > 0$ approaches a normally distributed random variable with mean μ and standard deviation $\sigma/\sqrt{n}$ in the limit of a large n. The theorem also applies to non-identically distributed random variables if

$$\lim_{n\to\infty}\left[\sqrt{\sum_{i=1}^{n}\sigma_{x_i}^{2}}^{\,2+\delta}\right]\sum_{i=1}^{n}E\left[|x_i - E(x_i)|^{2+\delta}\right] = 0 \quad \delta > 0$$

- Weak Law of Large Numbers -- The mean of a sequence of n random variables with the same mean μ and finite σ tends in the limit of large n to μ.

$$\lim_{n\to\infty} P\left[\frac{1}{n}\sum_{i=1}^{n}x_i = \mu\right] = 1$$

- Chebychev's Inequality –

$$\Pr\left[|x - E(x)| \geq h\sigma_x\right] \leq \frac{1}{h^2}$$

Information Theory

- Entropy – The *entropy* $H(X)$ of a discrete random variable X with alphabet $\mathcal{I}$ and probability mass function $p(x)$ is defined by

$$H(X) = -\sum_{x\in I} p(x)\log p(x)$$

- Differential Entropy – The *differential entropy* $h(X)$ of a continuous random variable with probability density function $p(x)$ and a support set S is defined as

$$h(x) \equiv -\int_S p(x)\log p(x)dx$$

- Joint Differential Entropy -- The *differential entropy* of a set of continuous random variables $X_1, X_2, \ldots, X_n$ with probability density function $p(x_1, x_2, \ldots, x_n)$ is defined as

$$h(x_1, x_2, \ldots x_n) \equiv -\int p(x_1, x_2, \ldots x_n) \log p(x_1, x_2, \ldots x_n) dx_1 dx_2 \ldots dx_n$$

- Conditional Differential Entropy -- The *conditional differential entropy* $h(X|Y)$ of a two continuous random variables with joint probability density function $p(x,y)$ is defined as

$$h(X|Y) \equiv -\int p(x,y) \log \frac{p(x,y)}{p(y)} dx$$

- Mutual Information – The mutual information between two random variables with joint density $f(x,y)$ is defined as

$$I(X;Y) \equiv -\int p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dx\, dy$$

- Kullback Liebler Distance – The *Kullback Liebler Distance* or *relative entropy* between two densities f and g is defined by

$$D(f\|g) = \int f \log \frac{f}{g}$$

Linear Algebra / Linear Programming

- A set C of vectors is *convex* if it satisfies

$$\text{if } x,y \in C \text{ and } 0 \le \lambda \le 1, \text{ then } \lambda x + (1-\lambda) y \in C.$$

- The *convex hull* of a set X of vectors is the smallest convex set containing X, and is denoted by conv.hull X; so

$$\text{conv.hull } X = \left\{ \lambda_1 x_1 + \ldots + \lambda_t x_t \middle| t \ge 1; x_1, \ldots, x_t \in X; \lambda_1, \ldots, \lambda_t \ge 0; \lambda_1 + \ldots + \lambda_t = 1 \right\}$$

- A *convex cone* is a non-empty set of vectors C satisfying

$$\text{if } x,y \in C \text{ and } \lambda, \mu \ge 0, \text{ then } \lambda x + \mu y \in C.$$

Dimensioning and Tolerancing

The following definitions are taken from ANSI Y14.5M:

- Feature. The general term applied to a physical portion of a part, such as a surface, hole, or slot.
- Dimension. A numerical value expressed in appropriate units of measure … to define the size or geometric characteristic, or both, of a part or feature.
- Basic Dimension. A numerical value used to define the theoretically exact size, profile, orientation, or location of a feature or datum target. It is the basis from which permissible variations are established by tolerances on other dimensions…
- High Limit. The value of a dimension above which the part will be considered as a defect.
- Low Limit. The value of a dimension below which the part will be considered as a defect.
- Tolerance. The total amount by which a specified dimension is permitted to vary. The tolerance is the difference between the high and low limits.
- Form Tolerance. A tolerance defining a zone in which all the points on the surface of a part must lie. Form tolerances include straightness, flatness, circularity, cylindricity, and occasionally profile. Form tolerance are applicable to single (individual) features or elements of single features; therefore, form tolerances are not related to datums.
- Dimension of Form. (not defined in ANSI) – A dimension of a part whose value equals the tightest tolerance of form which a given part would meet. ????
- True Position. The theoretically exact location of a feature established by basic dimensions.
- Datum. A theoretically exact point, axis, or plane derived from the true geometric counterpoint of a specified datum feature. A datum is the origin from which the location or geometric characteristics of features of a part are established.
- Datum Target. A specified point, line , or area on a part used to establish a datum.

Quality Control

The following definitions are from Juran's quality control handbook:

- Manufacturing system. A unique combination of machine tools, methods, materials, and people engaged in production.
- Tolerance width (TW). The total range over which the dimensions of the product may vary. This is identical to the term "tolerance" as defined in section 4.1 above.

- Process capability index (C_p). The ratio of the tolerance width to the process specification width. The process width is defined as six times the standard deviation of the dimension of a population produced by the process.

$$C_p = \frac{UT - LT}{6\sigma}$$

- Performance index (C_{pk}). A measure of process capability that attempts to account for non-centering of the process relative to its specification limits.

$$C_{pk} = \left(1 - \frac{|\bar{X} - D|}{S/2}\right) \cdot C_p$$

Signals and Systems

- A *signal* is a vector of physical quantities each of which is a function of one or more independent variables.
- A *system* is any process that results in the transformation of signals [Oppenheim and Willsky, 1983]. The system can therefore be viewed as a vector transform **T** from an input vector **x** into a response vector **y**.

$$\mathbf{T}(\mathbf{x}) = \mathbf{y}$$

- A *discrete time system* is one that transforms discrete time inputs into discrete time outputs [Oppenheim and Willsky, 1983].
- A system is *memoryless* if its response at a given time is dependent only on the signal at the same time.
- A system is *time invariant* if a time shift in the input signal causes a shift in the output signal.
- A system is said to be *invertible* if distinct outputs lead to distinct inputs.
- A system with the property of superposition is *linear*. A system has the property of superposition if it has both the *scaling property* and the *additivity property* which can be expressed mathematically as (respectively):

$$\mathbf{T}(a\mathbf{x}) = a\mathbf{T}(\mathbf{x})$$

$$\mathbf{T}(\mathbf{a} + \mathbf{b}) = \mathbf{T}(\mathbf{a}) + \mathbf{T}(\mathbf{b})$$

Random Processes

- Random Process -- An extension of a random variable to include variation over time. If a function of time $x(t,s)$ is assigned a value for every outcome s then the ensemble of the functions $x(t,s)$ for all outcomes s is a random process and is denoted $X(t,s)$. A random process becomes a random variable when time is fixed at some given value.
- Continuous Random Sequence – A random process for which $X(t,s)$ is continuous but t is discrete.
- Determinism – A random process is deterministic if future values of any sample function $x(t,s)$ can be determined from past values.
- Stationarity – A random process is said to be stationary if all its statistical properties do not change with time.

B. Lead Forming Analysis

This appendix documents preliminary efforts to model the forming of flat leads on electronic components. It focuses on the effects of misalignment of electronic components in the dies and its effect on lead geometry and conformance to MIL-STD-2000A.

PROBLEM DESCRIPTION

Many of the electronic components arriving at production facilities have unformed ribbon leads. The leads are therefore formed in-house on an automated die system. Malformation of the leads is sometimes detected downstream in the assembly process either by assembly personnel or by automated visual inspection systems. The malformation of the leads is sometimes highly localized suggesting the leads were damaged during handling. In other cases the malformation exhibits a regular pattern across the leads suggesting that the cause may lie in the purchased component or in the lead forming process. We wish to consider possible sources of this type of regular, patterned lead malformation.

The lead forming dies themselves are generally built to very high tolerances suggesting that the form of the dies is not a likely cause of variation. However, in some cases, sufficient means does not currently exist for alignment of the electronic components in lead forming dies. This analysis is an attempt to explore the possibility that certain lead malformations may be caused by misalignment of the electronic components in the dies.

DEFINING THE COMPONENT GEOMETRY

The model requires as input a description of the MCM package including body and lead dimensions. The terms employed in the model are given in Figures A.1 and A.2 below.

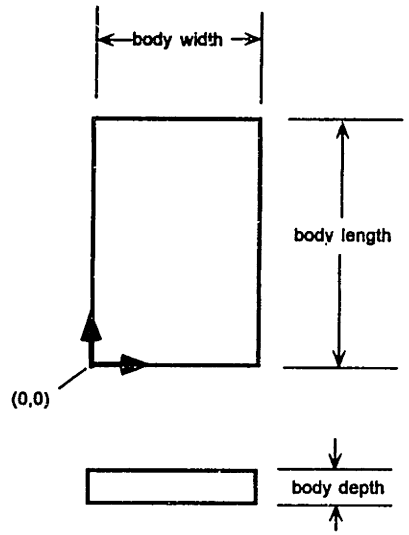


Figure B.1 -- MCM body dimensions.

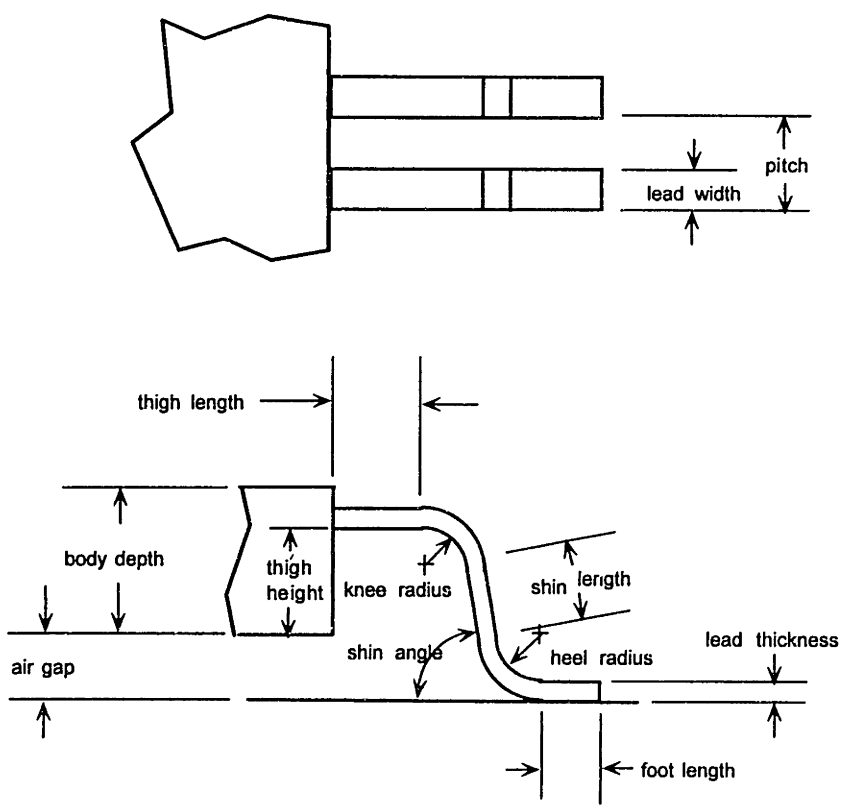


Figure B.2 Lead dimension terminology.

DEFINING ALIGNMENT ERRORS

Several alignment errors of the MCM in the lead forming die are possible. One of particular concern is depicted in Figure 3. There is some variation in the dimensions of the MCM bodies, especially where the same parts are provided by more than one supplier. Therefore, there may be

some clearance between the MCM body and the features on the die that are designed to support the corners of the MCM body. This may allow the MCM body to yaw slightly (Fig. B.3). The simulation calculates the maximum misalignment of the MCM automatically. However, several orientations of the chip are possible under these conditions (e.g. the chip can rotate in either direction and may have room to slide to one side or the other) and it is incumbent on the analyst to choose which to consider for a given simulation run.

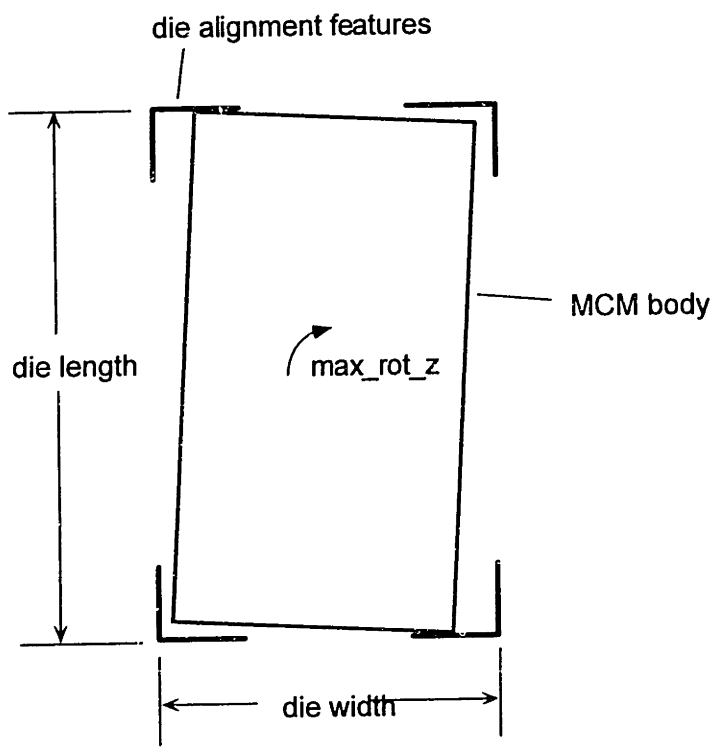


Figure B.3 Rotational error motion about the z axis.

Another possible errors are pitch or roll of the MCM in the die, misalignment of the leads with the MCM body, and misalignment of the MCM with the pads due to inaccuracies of the Adept robot.

MODEL OF THE LEAD FORMING PROCESS

An efficient model the lead forming process requires a compact representation of the unformed leads and the features formed from those leads. In the simulation, the unformed leads are

represented by a four by four matrix. Column vectors of that matrix are the homogeneous coordinates of the four corners of the center plane of the unformed lead.

Figure B.4 graphically represents the general flow of the simulation. First the leads are perturbed according to the misalignments and other error motions defined by the analyst. Then the leads are set flush to the surface of the lower half of the die. The simulation finds the points of intersection between the sides of the unformed leads and the projection of the edge at the beginning of the knee radius onto the center plane of the lead (Fig B.4 a). Those two points define a cutting line where the unformed lead is divided into two new entities. One is the thigh of the lead, the other is the remainder of the unformed lead.

The remainder of the unformed lead is then rotated about the axis of the knee radius and simultaneously translated along that axis (Fig B.4 b). The rotations and rotations simulate the skewed wrapping of the lead around the die knee bend. The remainder of the unformed lead is then intersected with the beginning edge of the heel radius. The unformed lead is cut to create the shin.

Just as when forming the knee, the remainder of the unformed lead is rotated about the axis of the heel radius and simultaneously translated (Fig B.4 c). The remainder of the unformed lead is then intersected with the edge of the die. The unformed lead is then trimmed to create the foot.

The resulting thigh, shin, and foot are collections of four points. In Figure 4 d they are in the same position that they would be while the upper half of the die is pressing down upon them. Once the upper half of the die is released, the foot and thigh will spring back to some extent. This spring back is given by the formula [Kalpakiian, 1995]

$$\frac{R_i}{R_f} = 4\left(\frac{R_i Y}{ET}\right)^3 - 3\left(\frac{R_i Y}{ET}\right) + 1. \tag{B.1}$$

This formula can be used to define transformations that simulate the spring back of the heel and shin. The final result is a simple representation of the formed lead. No explicit description of the geometry of the heel and knee is stored by the simulation, but the shape can readily be inferred from the thigh, shin, and foot geometry.

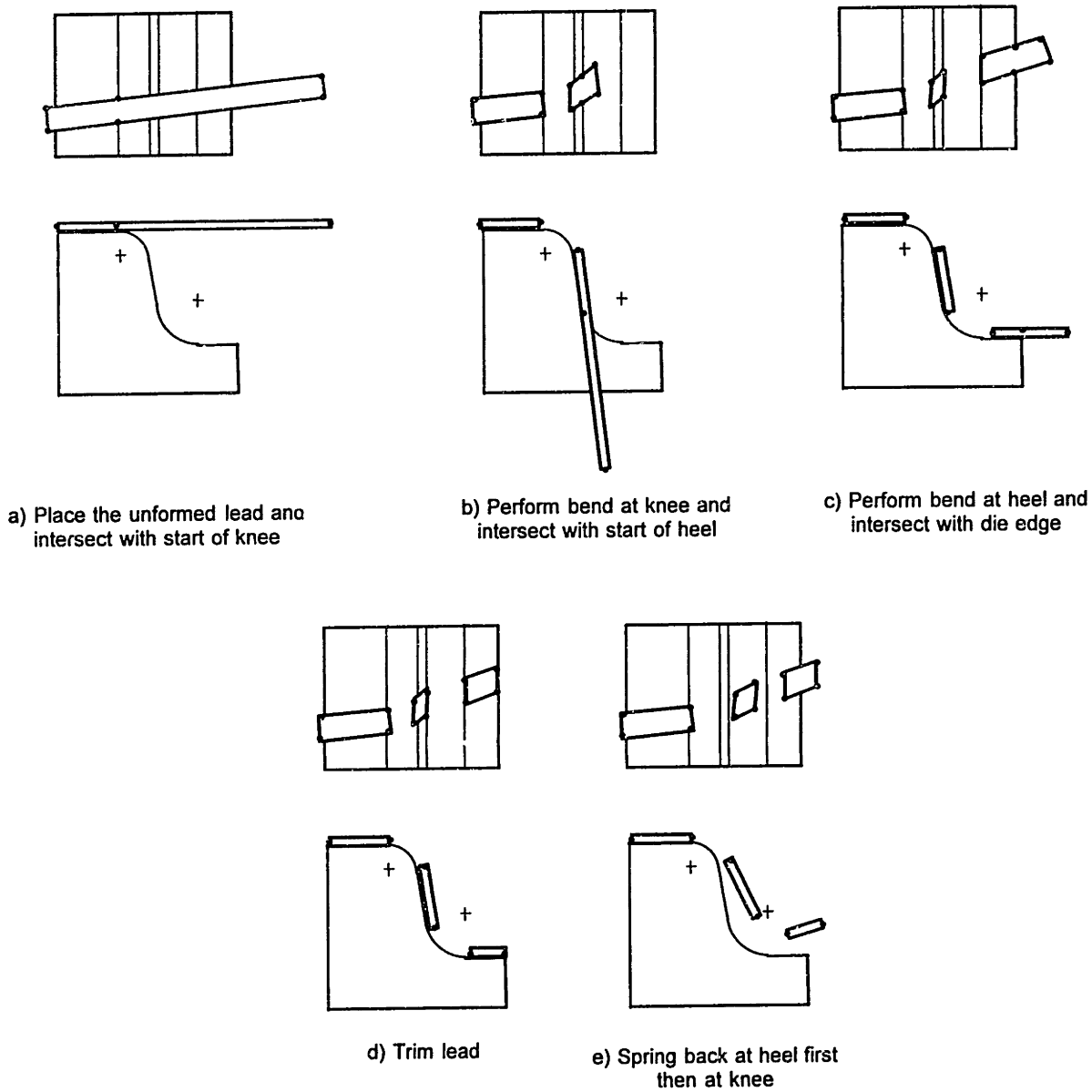


Figure B.4 The method used to model the lead forming operation.

LEAD INSPECTION BY MIL-STD-2000A

The Military Standard – Standard Requirements for Soldered Electrical and Electronic Components, MIL-STD-2000A establishes requirements for materials and procedures for making soldered electrical connections. It contains several items which directly apply to the formed shape of leads and/or their placement with respect to the solder pads. The model simulates the

placement of the MCM with formed leads onto a PWA and inspects the leads/pad arrangement according to the following specifications:

4.23.7.1 Surface mounted device lead bend configuration. Lead bends shall not extend into part body lead seal.

This test is performed by ensuring that none of the points on the thigh extend into the MCM body.

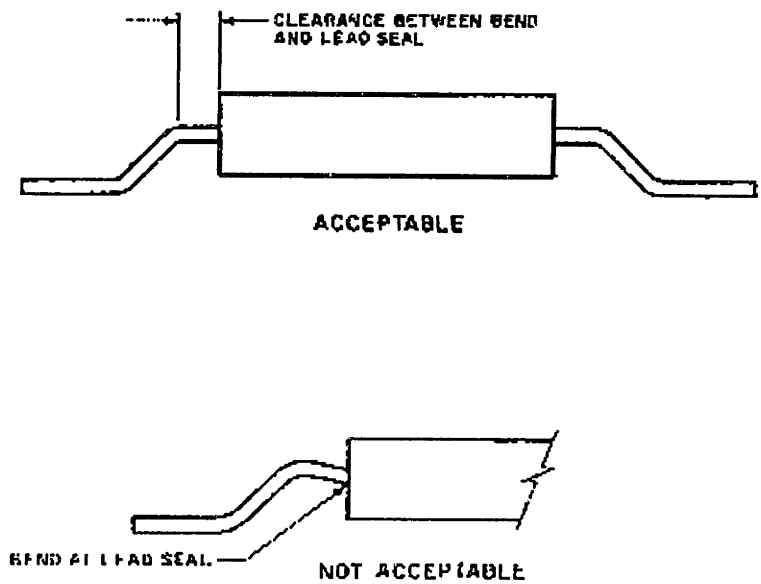


Figure B.5 Lead bends cannot extend back to MCM body.

4.23.7.2 Surface mounted device lead and land contact. ... Side overhang is permissible provided it does not exceed 25 percent of the lead width or... 0.02 inch, whichever is greater

This test is performed by checking the positions of the four corners of the foot with respect to the four corners of the pad.

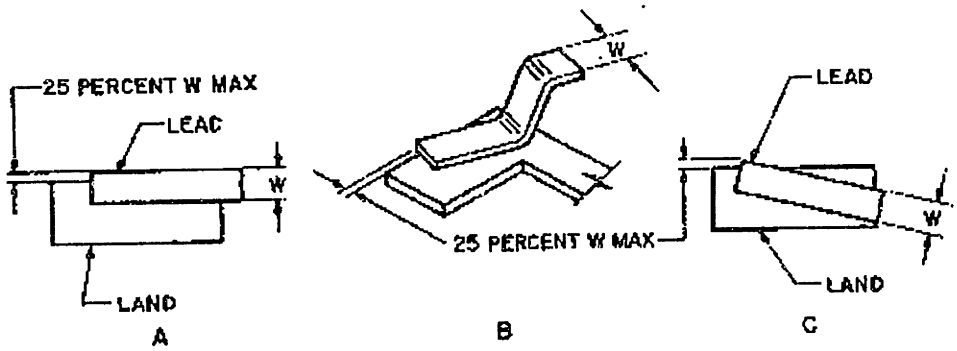


Figure B.6 The foot cannot extend over the pad side excessively.

4.23.7.2 Surface mounted device lead and land contact. Toe end overhang is permissible provided the total overhang does not exceed 25 percent of the lead width ... or 0.02 inch, whichever is less...

This test is performed by checking the position of the two most distal corners of the foot with respect to the corresponding corners of the pad.

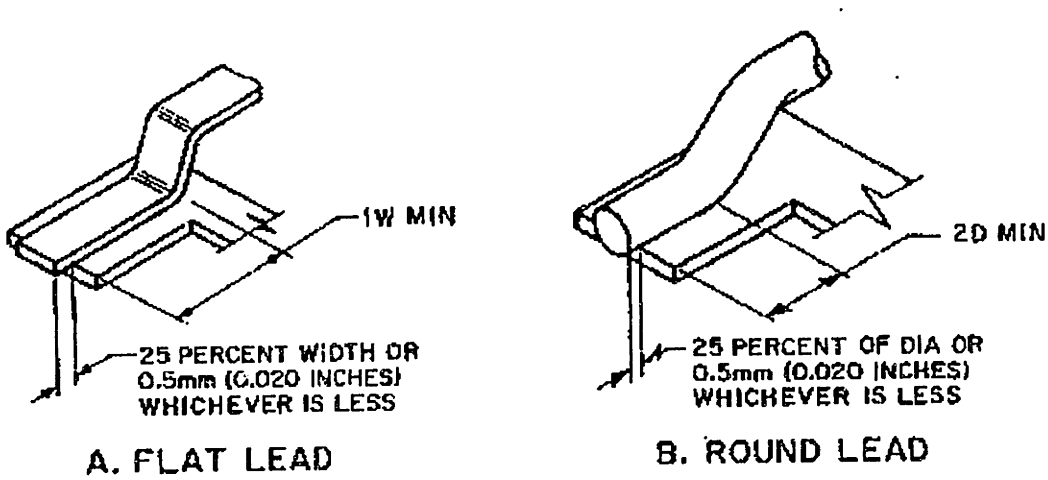


Figure B.7 The foot cannot extend outboard of the pad excessively.

4.23.7.2 Surface mounted device lead and land contact. The heel shall not overhang the land...

This test is performed by checking the two distal corners of the shin with respect to the inboard corners of the pad.

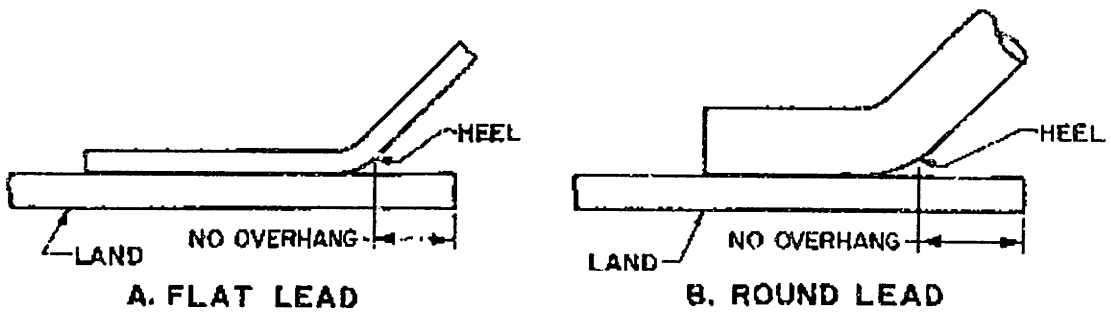


Figure B.8 The heel cannot overhang the pad.

4.23.7.3 Surface mounted device lead height off land. Flat or ribbon leads may have their minimum seating plane raised off the land surface a maximum of two times the lead thickness or ... 0.02 inch, whichever is less. All points within the minimum seating plane shall be within ...two lead thickness maximum spacing... Toe up or toe down... shall be permissible provided that separation between the leads and termination area does not exceed 2T ...

This test is performed by measuring the angle formed by the sides of the toe with the horizontal plane.

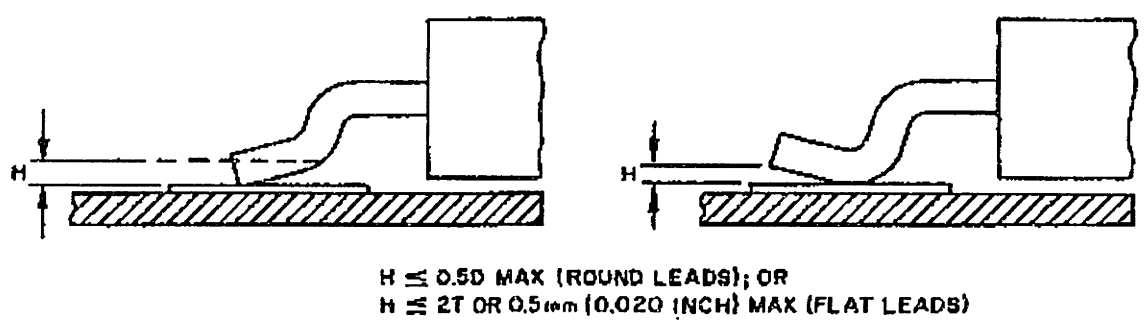


Figure B.9 The foot cannot toe up or down excessively.

The simulation provides indication of one for pass and zero for fail for each lead for each test, a composite pass/fail indication for each lead, and a final pass/fail indication for the MCM and placement as a whole.

ALIGNMENT OF THE LEADS WITH THE PADS

The model simulates the automatic alignment procedure performed by an assembly robot. First, the model computes the centriods of the feet and the pads. Then it computes a good starting point for the least squares fit with a "common sense" procedure. Then a non-linear solver is used to find the x translation, y translation, and rotation about the z axis that minimizes the sum of the squares of the distances from the centriods of the feet to the centriods of the corresponding pads.

RESULTS

Some preliminary simulation runs have been performed to get a sense of the model's behavior and the type of results it may be able to provide. For example, Figures B.10 through B.14 depict the lead arrangement when the MCM is misaligned in yaw with the die (as in Figure B.3) by 1.4 degrees. This causes the leads to drift up on the left side and down on the right.

MCM Package - Top view

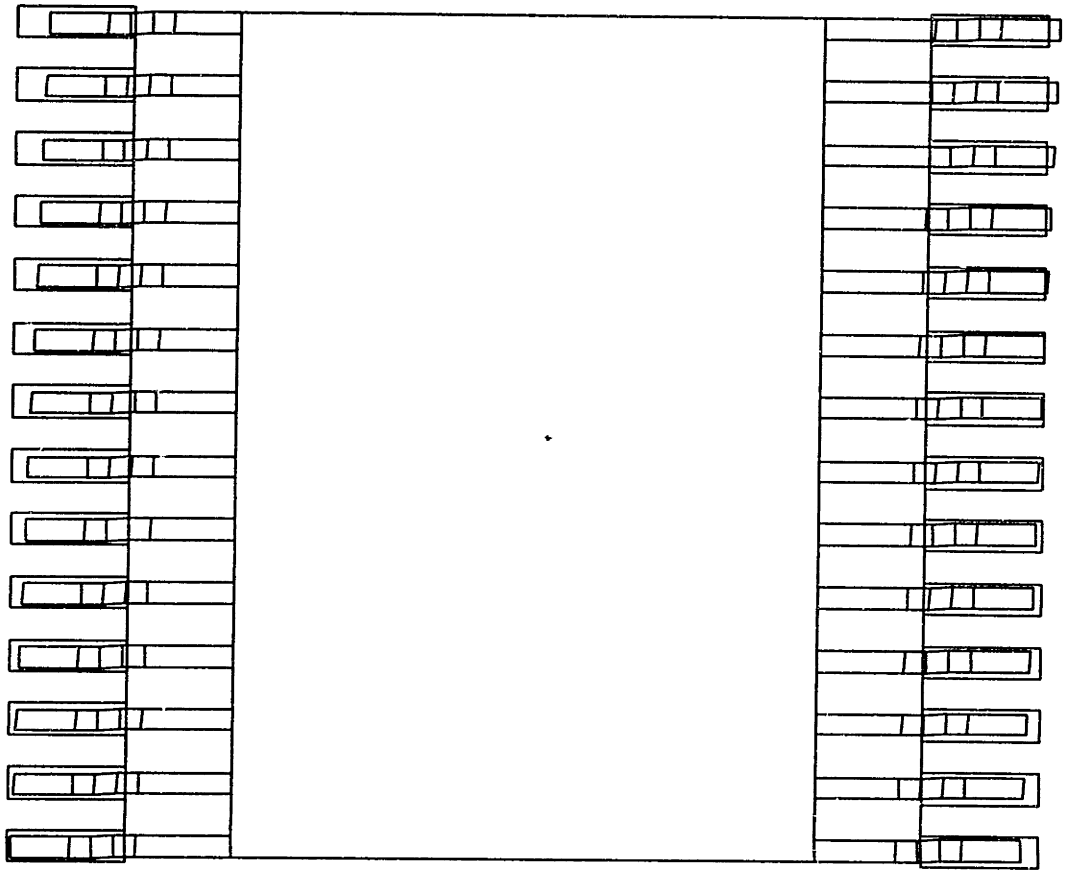


Figure B.10 Lead pattern due to yaw of the MCM in the die - top view.

MCM Package - Front View

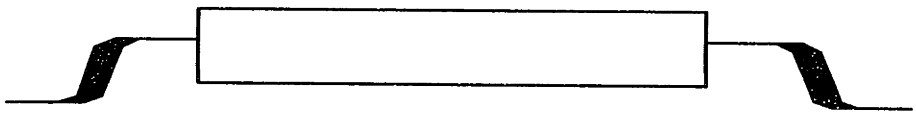


Figure B.11 Lead pattern due to yaw of the MCM in the die - front view.

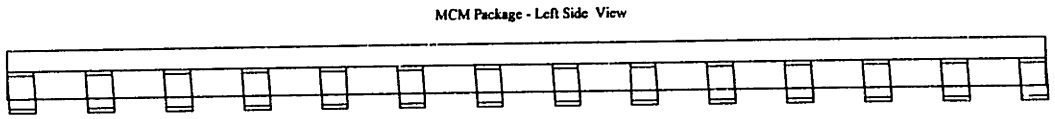


Figure B.12 Lead pattern due to yaw of the MCM in the die - right side view.

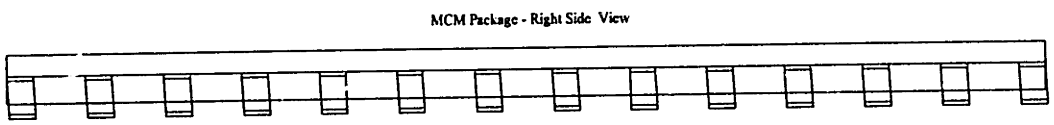


Figure B.13 Lead pattern due to yaw of the MCM in the die - right side view.

CONCLUSIONS

These preliminary simulation runs do suggest that patterns of error in the leads can result from misalignment of the MCM with the die. It also seems that simple simulations may be effective in revealing both the magnitude and pattern of the resulting error.

C. Adhesive Flow Analysis

This appendix documents efforts to model adhesive flow between a Multichip Module and a Printed Wiring Assembly. The numerical results presented do not use genuine measurements as input and are only provided to indicate trends in the model behaviors. Closed form expressions are developed for the MCM height versus time behavior of a long bead of adhesive and of a circular pattern of adhesive. These results are used to estimate the behavior of rectangular patterns of adhesive. Methods for modeling the behavior of adhesive patterns with entrapped air pockets are introduced. Preliminary results of a numerical solutions of the model are presented. They suggest that air pockets add substantially resist downward motion of the MCM and tend cause "spring back" of the MCM when the applied force is relaxed.

PROBLEM DESCRIPTION

In many electronic package designs, a layer of adhesive is placed between a Printed Wiring Assembly (PWA) and a Multichip Module (MCM). The layer serves to fix the MCM to the PWA, conduct heat from the MCM body into the PWA, and support the MCM during shocks and vibrations. In the process under consideration, the adhesive is a viscous, thixotropic liquid which becomes an elastomeric solid upon curing.

Often the adhesive is applied to the PWA by a robot. The adhesive is contained in a plastic tube with a conical tip. The adhesive is forced from the tube by air pressure. The robot lays adhesive beads in a "plow" pattern such as that shown in Figure 1. The plow pattern is typically framed with a bead of adhesive around perimeter. This seals and protects the traces on the PWB under the MCM from liquids.

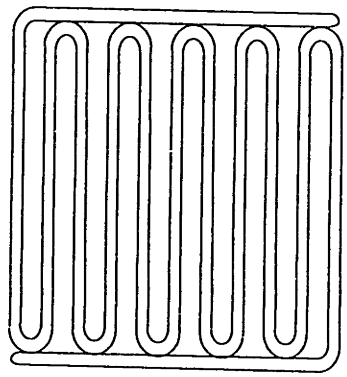


Figure C.1 A pattern of adhesive laid down by a robot.

Some possible problems that may be encountered with adhesive application include:

- The adhesive may fail to bond adequately to the MCM package. If there is not adequate surface area of contact among the MCM, adhesive, and PWA, then the MCM will not be cooled sufficiently. This may be caused by inadequate thickness of adhesive beads, excessively viscous adhesive, excessively long shins on the leads, inadequate downward force applied to the MCM, inadequate time of force application, waviness of the PWA, or possibly other causes.
- The adhesive may flow out from under the MCM body, making contact with the leads. This is a serious concern because the adhesive may contaminate the solder joint. This may be caused by excessive thickness of adhesive beads, low viscosity of the adhesive, shorter than expected shins on the leads, excessive downward force applied to the MCM, excessive time of force application, inadequate clearance from the bead pattern to the edge of the MCM, or possibly other causes.
- The outer perimeter of the bead pattern may burst due to air pressure beneath the MCM. This is a serious concern because the resulting gap under the MCM body can allow liquids to be trapped under the MCM during later processing or in service. Liquids trapped under the MCM could corrode the traces of the PWA causing the package to fail in service.

These production problems motivate an effort to model the manufacturing process. It is hoped that a model will help process engineers to:

- Estimate the effects of variation of process variables (e.g. lot to lot variations in adhesive viscosity) on product quality.
- Test out new designs for adhesive bead patterns, force versus time schedules, etc.
- Reevaluate specifications on PWA flatness, adhesive viscosity, and other variables.

MODEL OF A LONG BEAD

One goal is to understand the important characteristics of the flow of the adhesive as it is compressed between the MCM and PWA. As a first step, I considered the flow of a single, long bead in isolation; That is, I considered the behavior of a bead free from the influence of the neighboring beads.

The key to this and subsequent models is the assumption of locally fully developed (LFD) viscous flow. LFD flow theory begins with the Navier-Stokes equations of flow of a fluid with constant viscosity. This assumption may have to be amended later if the fluid proves to have a viscosity that is substantially dependent on shear rate. We suspect that this is the case, but considering the simpler flow first may still prove illuminating. It may also be possible to define an apparent or average viscosity for the adhesive so that the equations and models developed with the assumption of constant viscosity can be employed to estimate the behavior of the system.

The main feature of LFD flows that simplify their analysis is that both temporal and convective inertial forces are negligibly small compared to viscous forces. In this context, the assumptions are:

- The local angle of divergence between the MCM body lower face and the PWA must be everywhere small.
- The characteristic time associated with significant velocity change is long compared with the time h^2/ν where h is the vertical distance from the MCM body to the PWA. This criteria does not rule out analyses of problems with time dependent boundary conditions such as the present problem.
- The product of the Reynold's number of the flow and the ratio h/L is much less than one. The ratio h/L is approximately the angle between the MCM body and the PWA.

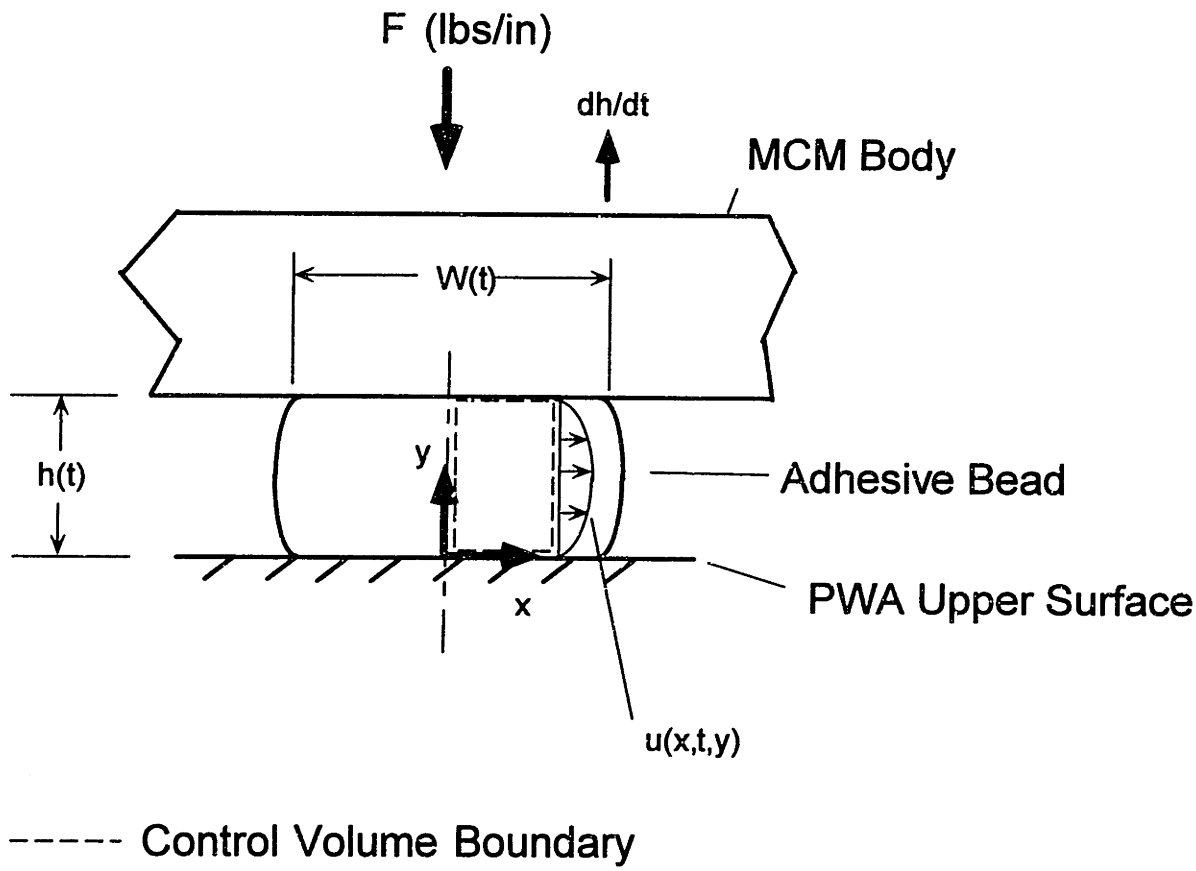


Figure C.2 Notation for the model of a single adhesive bead

Another key simplifying assumption is that the bead is long compared to its width. This implies that the flow is two dimensional. The flow along the length of the bead will be small compared to the flow in the width direction.

If all of the above conditions hold then, applying the no slip boundary condition at the wall of the MCM and PWB, we find that the flow pattern at any point x is given by:

$$u(x,t,y) = \frac{-h(x,t)^2}{2\mu}\frac{y}{h(x,t)}(1-\frac{y}{h(x,t)})\cdot\frac{dp}{dx} \tag{C.1}$$

This is the parabolic profile characteristic of Poiseuille flow. In fact, this is a form of Poiseuille flow with time dependent boundary conditions. To begin with, I will assume that the MCM body and PWA are flat and parallel so that h is a function of time only and not of x position. The total flow of material past a vertical line at x is therefore

$$\int_0^h u(x,t,h) = \frac{-h^3}{12\mu} \cdot \frac{\mathrm{d}p}{\mathrm{d}x} . \tag{C.2}$$

Applying conservation of mass to a control volume of adhesive (whose density is assumed constant under the pressure changes in this problem) running from the bead centerline to x and combining with the equation above yields

$$-x \cdot \frac{\mathrm{d}h}{\mathrm{d}t} = \frac{-h^3}{12\mu} \cdot \frac{\mathrm{d}p}{\mathrm{d}x} \tag{C.3}$$

Rearranging and integrating, applying the boundary condition that pressure in the adhesive equals atmospheric pressure at x=W/2 we obtain

$$p(x) = \frac{6\mu}{h^3} \cdot \frac{\mathrm{d}h}{\mathrm{d}t} \cdot (x^2 - (W/2)^2) . \tag{C.4}$$

So there exists a parabolic pressure distribution in the bead with the greatest pressure at the center of the bead. This pressure supports the downward force on the MCM so that

$$\frac{F}{L} = 2 \int_0^{L/2} p(x,t)\mathrm{d}x = \frac{-\mu}{h^3} \cdot \frac{dh}{\mathrm{d}t} \cdot W^3 \tag{C.5}$$

where F/L is the downward force applied per unit length of bead. Here the weight of the MCM is either neglected or subsumed in F/L. From conservation of mass, we know that the product of the width times the height of the bead is constant over time. Here it is convenient to assume that the bead is roughly rectangular in cross section. The bead will, in fact, likely have a rounded appearance at the ends as in Figure 1. Combining the conservation of mass and Equation 5 above, rearranging, and integrating with the boundary conditions $t = 0$, $h = h_o$ and $t = t_o$ yields

$$h(t) = \left[\frac{5F/L}{\mu W_o^3 h_o^3} t + \frac{1}{h_o^5} \right]^{-1/5} . \tag{C.6}$$

To give some sense of the behavior of this equation it is plotted in Figure 3 below assuming $h_o = 50mils$, $W_o = 0.2in$, $\frac{F}{L} = 0.2\frac{lbf}{in}$, and $\mu = 10\frac{lbf \cdot \sec}{in^2}$

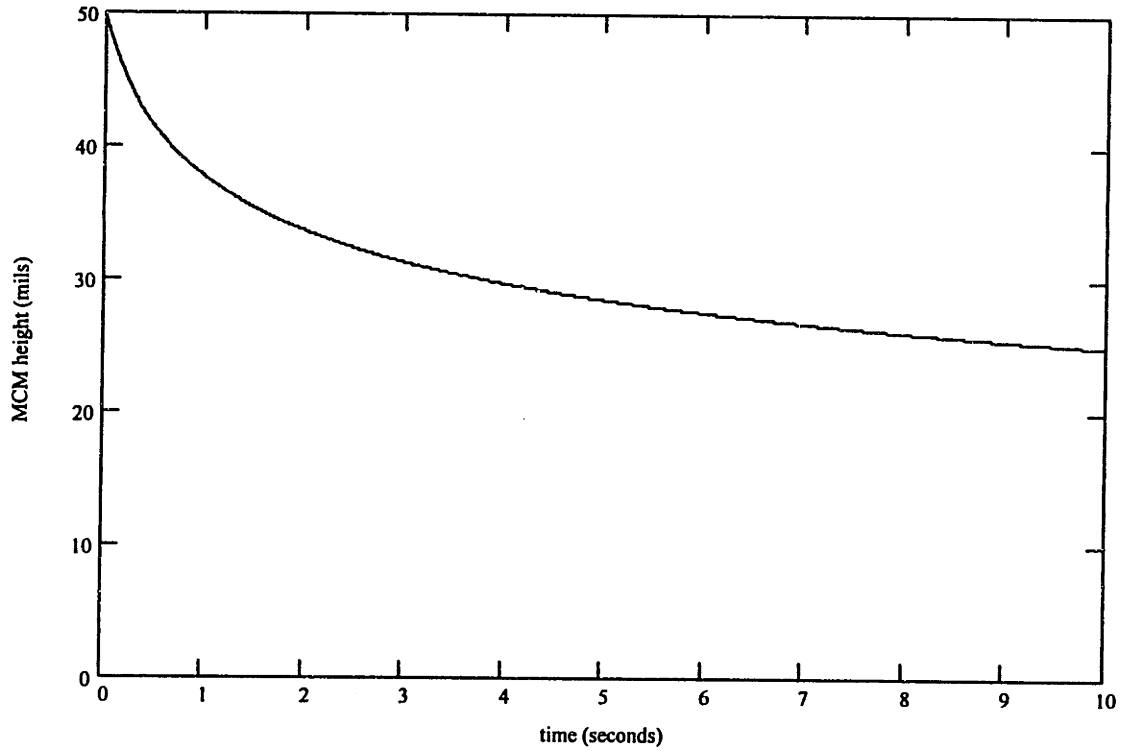


Figure C.3 Air gap height versus time for a single adhesive bead.

The MCM is pressed down rapidly at first then settles slowly toward the PWA. The increased resistance to downward motion is due to two factors. As the gap between the MCM and PWA decreases, the outboard flow of adhesive is restricted. Also, as the bead widens, there is a rise in the area over which the pressure in the adhesive acts to counter the downward force on the MCM.

MODEL OF A CIRCLULAR PATTERN

In some cases, glue beads may be laid down next to one another with no gap in between to create a continuous pattern of glue. In practice, these patterns will likely be rectangular with rounded corners. Analysis of these rectangular patterns is difficult, so it is instructive to first analyze a *circular* layer of adhesive (see Figure 4).

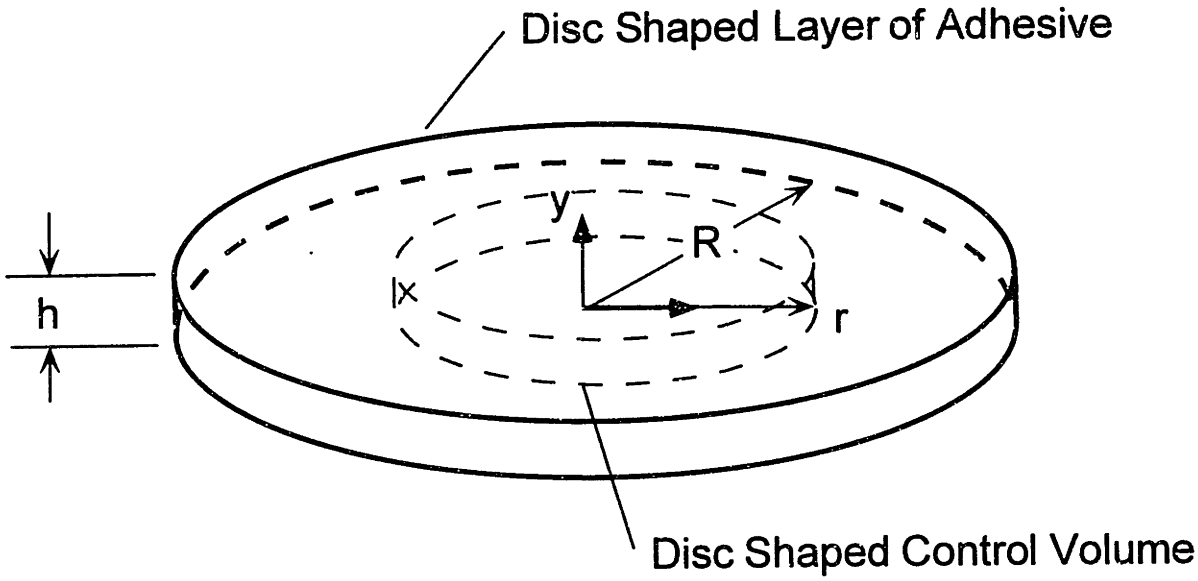


Figure C.4 Notation for a circular layer of adhesive.

In a disc of glue subject to approach of parallel walls, the flow pattern will be radially symmetrical. This analysis will therefore be essentially two dimensional. Equation 2 from above will hold except that the pressure gradients are radial

$$\int_0^h u(r,t,h) = \frac{-h^3}{12\mu} \cdot \frac{\mathrm{d}p}{\mathrm{d}r}. \tag{C.7}$$

Mass conservation applied to a disc shaped control volume yields

$$-\pi r^2 \frac{\mathrm{d}h}{\mathrm{d}t} = 2\pi r\left(\frac{-h^3}{12\mu} \cdot \frac{\mathrm{d}p}{\mathrm{d}r}\right). \tag{C.8}$$

Rearranging and integrating, applying the boundary condition that pressure in the adhesive equals atmospheric pressure at $r=R$ we obtain

$$p(r) = \frac{3\mu}{h^3} \cdot \frac{\mathrm{d}h}{\mathrm{d}t} \cdot (r^2 - R^2) \ . \tag{C.9}$$

Again there exists a parabolic pressure distribution in the adhesive with the greatest pressure at the center. This pressure supports the downward force on the MCM so that

$$F = \int_0^R \int_0^{2\pi} p(r,t) r \mathrm{d}\theta \mathrm{d}r = \frac{-3\pi\mu}{2h^3} \cdot \frac{dh}{\mathrm{d}t} \cdot R^4 \tag{C.10}$$

From conservation of mass, we know that the volume of the glue disc is constant over time. Combining the conservation of mass and Equation 10 above, rearranging, and integrating with the boundary conditions $t = 0$, $h = h_o$ and $t = t_o$ yields

$$h(t) = \left[\frac{8F}{3\pi\mu R_o^{\,4} h_o^{\,2}} t + \frac{1}{h_o^{\,4}} \right]^{-1/4} . \tag{C.11}$$

The larger negative exponent in this equation 11 as compared to equation 6 reflects the lesser damping in a circular arrangement allowed by the radial as opposed to uni-directional flow of adhesive.

ESTIMATES FOR RECTANGULAR PATTERNS

Analysis of the flow of a rectangular pattern of adhesive is rather difficult. If the aspect ratio of the patterns is not significantly greater than one, as is most often the case, the assumption of unidirectional flow (used in the case of a long bead) will be inappropriate. An attempt to consider flow and pressure gradients in two directions did not yield closed form solutions. Also, the actual shape of the glue pattern will change as it is pressed down, bulging out at the centers of the edges. This makes the analysis even more difficult.

Although direct analysis of solid rectangular patterns of glue is difficult, it is simple to provide firm upper and lower bounds on the solution by considering the solution for a circular pattern.

An MCM supported by adhesive in a circle that *fits entirely within* a rectangle (Fig.4) will move down more than an MCM supported by a rectangle of adhesive if both are subject to the same downward force versus time schedule. The rectangle contains all the adhesive in the inscribed circle plus some extra that would have to impede the downward motion of the MCM. Thus analysis of this circle provides a lower bound on height of the MCM at any time.

An MCM supported by adhesive in a circle that has *the same area* as a rectangle (Fig. 4) will move down less than an MCM supported by a rectangle of adhesive if both are subject to the same downward force versus time schedule. The rectangle has the same amount of adhesive as the circle, but the circle is the most efficient possible shape for impeding downward motion of the MCM. Thus analysis of this circle provides an upper bound on height of the MCM at any time.

A second upper bound on the height is provided by Equation 6. Using an equation derived assuming zero flow in the length direction will clearly overestimate the damping of the system.

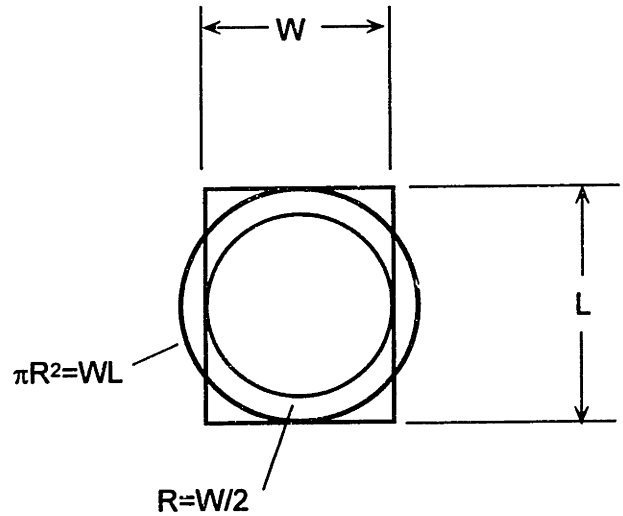


Figure C.3 Upper and lower bounds for a rectangular pattern

Figure C.5 is a plot of the downward motion of an MCM given $h_o = 50mils$, $W_o = 1in$, $L_o = 1in$, $F = 1lbf$, and $\mu = 10\frac{lbf \cdot \sec}{in^2}$. For a square pattern such as this one, the lowest upper bound and the lower bound are very close together (the hatched region in Figure 6 represents the area in which the exact solution may lie). Taking the mean of the upper and lower bounds gives an estimate with better than ±5% accuracy (assuming the other assumptions of the analysis hold).

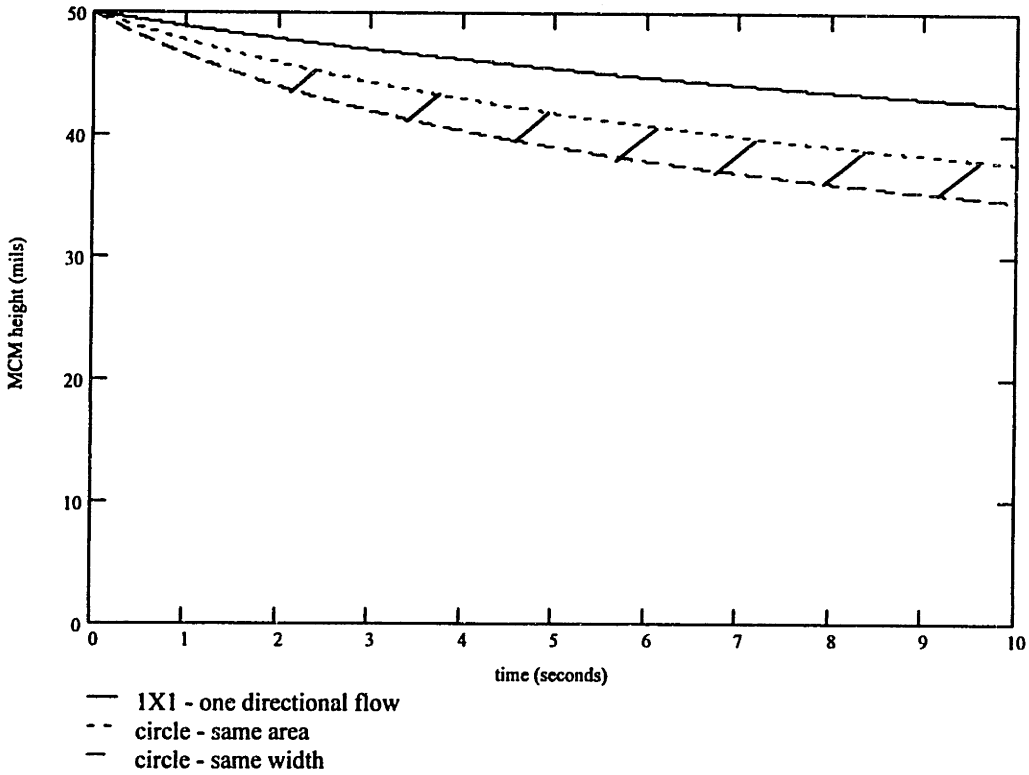


Figure C.4 Bounds on motion of a 1"X1" square pattern of adhesive.

Figure C.5 is a plot of the downward motion of an MCM given $h_o = 50mils$, $W_o = 1in$, $L_o = 2in$, $F = 1lbf$, and $\mu = 10\frac{lbf \cdot \sec}{in^2}$. For a pattern with an aspect ratio of two, the distance between the upper and lower bounds is larger. Intuitively, it is fair to say that the lower bound is clearly too low and that the upper bound is clearly too high. Therefore a mean of the two is a good estimate and the upper and lower bound provide a firm limits at about ±10%.

For an aspect ratio of two, the two upper bounds are very similar. For aspect ratios greater than two, the one directional upper bound estimate will be the active upper limit and as the aspect ratio approaches about ten it will be the best estimate of the height (the lower bound estimate will be too low to be of much use).

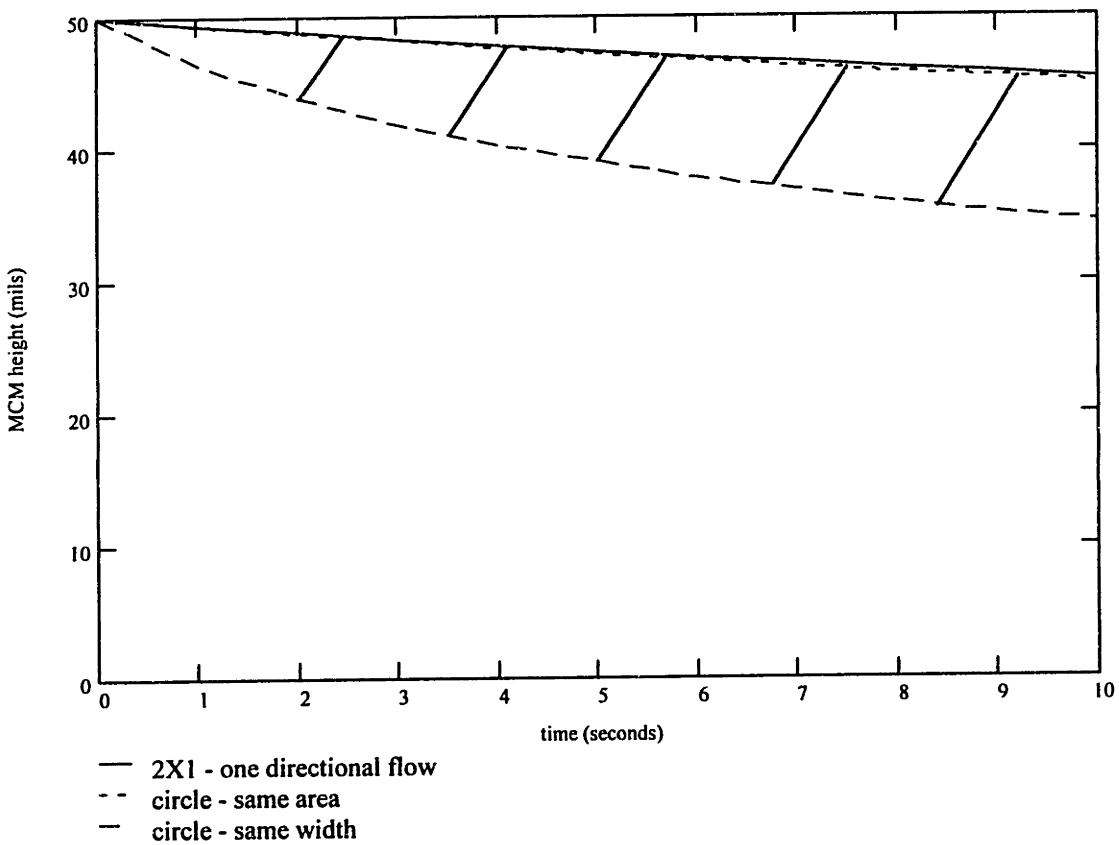


Figure C.5 Bounds on motion of a 2"X1" rectangular pattern of adhesive.

MODELING MULTIPLE BEADS WITH AIR POCKETS

If one employs a bead pattern similar to that in Figure 1, air pockets will be trapped between the beads. As the MCM is pressed toward the PWA, these air pockets will be compressed by the combined action of the downward motion of the MCM and the lateral expansion of the adhesive beads. This compression causes the pressure in the air to rise and the pressure will resist the

downward force on the MCM. One can think of this effect as adding highly nonlinear springs to a system of highly non-linear dash-pots.

For any given bead, the pressure in the adjacent air pocket on the outboard side is likely to be higher than the pressure in the pocket to the inboard side. This will cause the adhesive to flow outboard. This is not to say that the bead literally travels outboard. Adhesion to the MCM makes this seem unlikely. Rather, it is more likely that the inboard edge of the bead will propagate inboard, only more slowly than the outboard edge propagates outboard. Thus, although neither edge of the bead is receding, the centerline drifts outboard due to the pressure gradient. In analyzing this system, one must therefore consider the effects of each bead on its neighboring beads.

This system contains a large number of interdependent variables. In addition to the height of the MCM and the width of the beads, there are now the pressure in each pocket, and the position of each bead. A numerical solution seems to be necessary.

Figure 8 depicts a cross section of an MCM sitting on a pattern of beads with sealed air pockets such as that in Figure 1. Assume that the beads are spaced symmetrically about the center of the MCM and that there are an even number of beads. The vector $\mathbf{x}$ contains the instantaneous location of the centerline of each bead number consecutively from the center outboard as shown in Figure 8. The vector $\mathbf{p}_n$ is the air pressure in the gap adjacent to and inboard of the nth bead. Note that, with this notation, the total number of adhesive beads is $2n$.

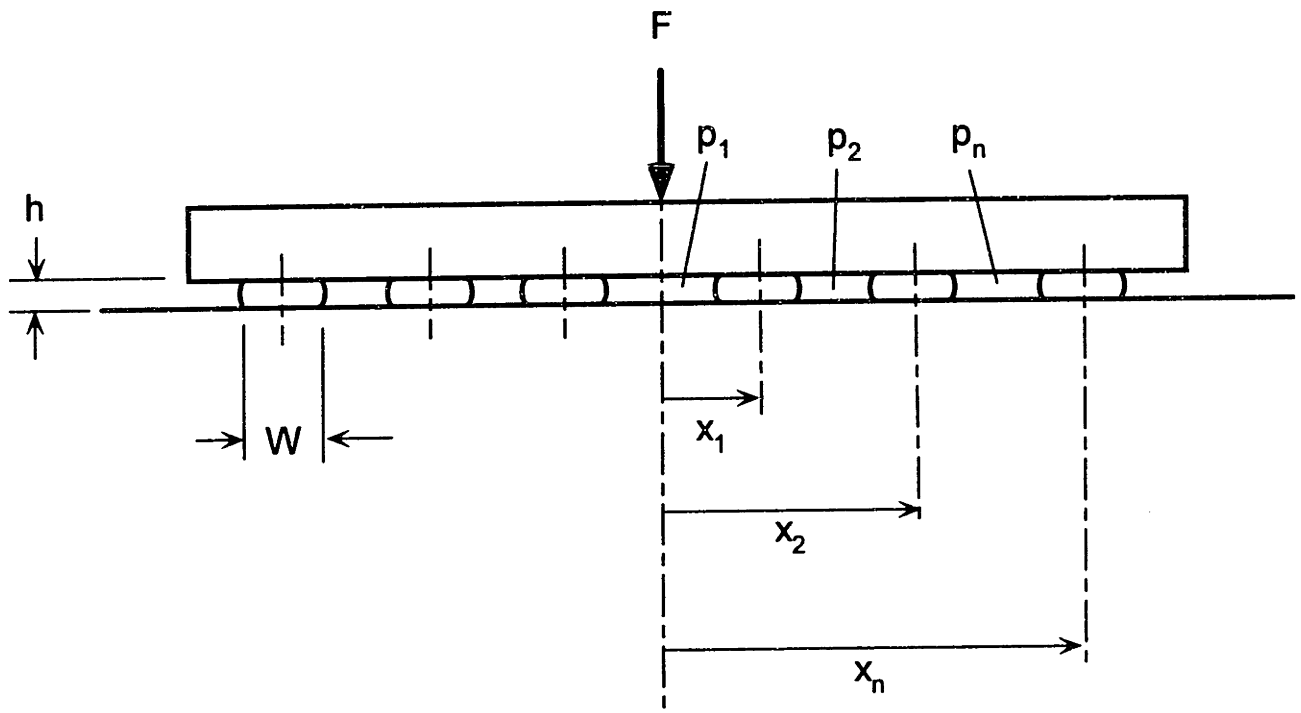


Figure C.6 Notation for a pattern with multiple adhesive beads with air pockets

Now I define the interdependence among the variables. The mass of each bead is conserved, therefore:

$$W(h) = \frac{W_o \cdot h_o}{h}. \tag{C.12}$$

The pressure in each bead is estimated by considering the air as an ideal gas. At the temperature and pressure one would expect to see in this application, this should be a reasonable approximation. The pressure of the air is therefore

$$\mathbf{p}_n = \begin{cases} p_a \cdot \dfrac{(2\mathbf{x}_{1_o} - W_o)h_0}{(2\mathbf{x}_1 - W)h} & \text{if } n = 1 \\ p_a \cdot \dfrac{(\mathbf{x}_{n_o} - \mathbf{x}_{n-1_o} - W_o)h_0}{(\mathbf{x}_n - \mathbf{x}_{n-1} - W)h} & \text{otherwise} \end{cases}. \tag{C.13}$$

These pressures cause a pressure gradient across the whole bead which in turn causes lateral movement of the beads. Since the flow of adhesive is everywhere a linear function of pressure gradient as indicated in Equation 1, one can consider this pressure gradient separate from that

due to downward motion of the plate and employ superposition later to combine the two effects. This effect will cause a constant pressure gradient across the bead proportional to the pressure difference and inversely proportional to W. This pressure gradient will cause an outboard velocity of the bead given by Equation 2. Therefore the rate of change of $\mathbf{x}$ is

$$\frac{d\mathbf{x}_n}{dt} = \frac{h^2}{12\mu} \cdot \frac{\mathbf{p}_n - \mathbf{p}_{n+1}}{W}. \tag{C.14}$$

The change in height of the beads with time is caused by whatever net force remains after the externally applied downward force is counteracted by upward pressure due to the pressure in the air pockets. The pressure profile across the lower face of the MCM due to this effect takes the shape of a Mayan pyramid with flat areas over the air pockets connected by linear pressure drops outboard through the adhesive beads. Integrating across the bottom of the MCM yields the net force

$$\frac{F_{net}}{L} = \frac{F}{L} - 2\cdot(\mathbf{p}_1 - p_a)(\mathbf{x}_1 - \frac{W}{2}) - 2\cdot\sum_{i=1}^{n}\left[W\cdot\left(\frac{\mathbf{p}_{n-1} + p_a}{2} - p_a\right) + (\mathbf{p}_n - p_a)\cdot(\mathbf{x}_n - \mathbf{x}_{n-1} - W)\right] \tag{C.15}$$

The rate of change of the height of the MCM is

$$\frac{dh}{dt} = -\frac{F_{net}}{L}\frac{h^3}{2\cdot \text{number\_of\_beads}\cdot \mu W^3}. \tag{C.16}$$

Equations 12-16 above define a system of first order ordinary differential equations. They can be solved numerically by Euler's method. An implementation of the solution in MathCad is attached as an Appendix.

PRELIMINARY RESULTS FOR MULTIPLE BEADS

To get a sense of the behavior of the model, we ran the simulation with a total of ten adhesive beads spaced $0.1in$ apart initially with $h_o = 50mils$, $W_o = 0.2in$, $L_o = 1in$, $F = 50lbf$, and $\mu = 10\frac{lbf\cdot\sec}{in^2}$. These values are for illustration only and do not reflect the actual capabilities of the production equipment.

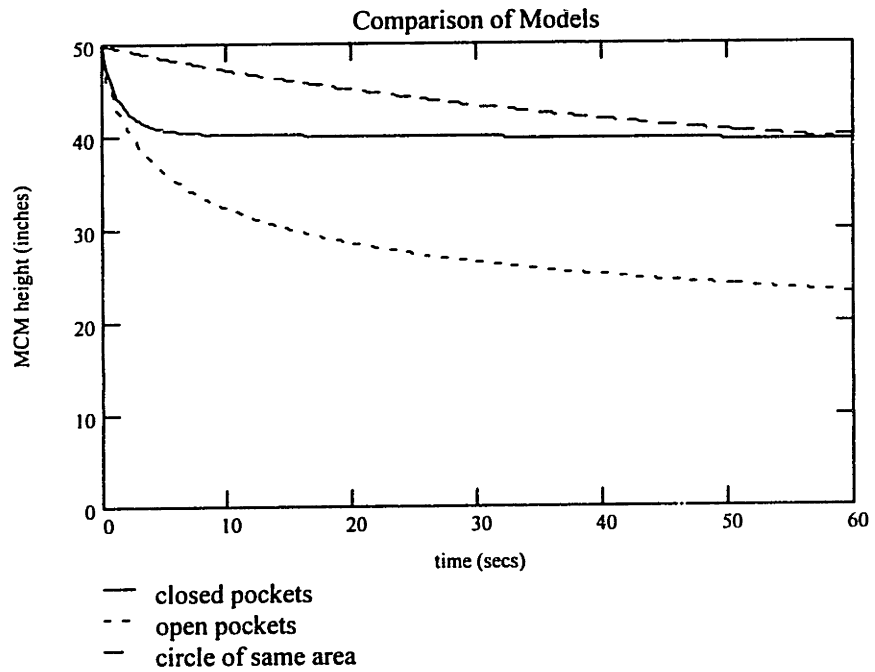


Figure C.7 Comparison of three different adhesive configurations.

Some of the simulation results are plotted in Figure 9 above. Also plotted are results for with a model assuming no pressure in the air pockets and large spacing between beads. Equation 6 was used with $F=5lbf$ (that assumes each bead shares the applied force equally). Observe that in the first moments of the simulation, while the air pressure in the pockets was low, the model behaved very much as if the pockets did not exist. However, once the pockets pressurized, their resistance to the downward force slows the downward motion very effectively.

Also plotted in Figure 9 is the behavior of a system with a circular disc of glue of the same area (area of glue only that is, not air *and* glue) as the simulated model with air pockets. Note that the pattern with pockets will resist downward force even more effectively than the solid circular pattern over an adequate time.

It is interesting to consider the behavior of the model after the applied force is relaxed. To explore this, we ran the model again with the same parameters as above, but with a modified

force versus time schedule as shown in Figure 10 below. Figure 11 shows the height versus time behavior of the model. Note that one can get the MCM to set down substantially under adequate force, but it will tend to spring back almost entirely when the force is removed. This is due to the fact that there is very little outboard motion of even the most outboard adhesive beads as shown in Figure 11. The lack of outboard adhesive flow is due, in turn, to the very flat pressure profile that evolves under the MCM as the force is held (see Figure 12). The adhesive beads near the center have very little pressure gradient across them and therefore do not flow outboard. The only way to achieve MCM downward motion that is not recovered upon force relaxation is through outboard adhesive flow. This is because the air pockets will force the MCM back up until they return to near zero gauge pressure (see Figure 13). Unless the air pocket pressure is relieved by outboard flow of adhesive, the MCM will spring back to its full original height.

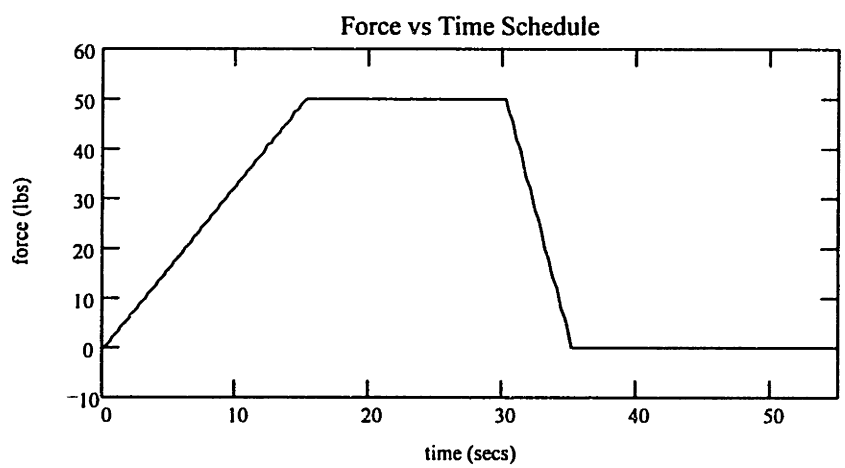


Figure C.8 Force versus time schedule for a simulation run.

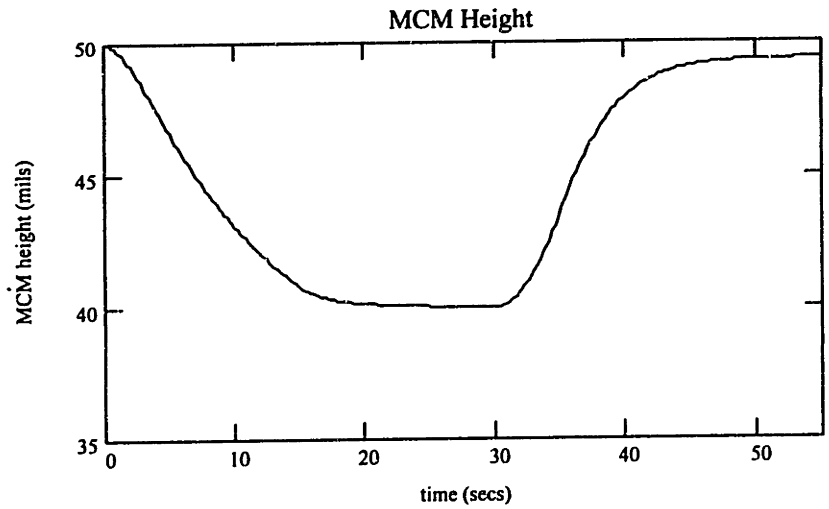


Figure C.9 MCM height versus time behavior including relaxed load.

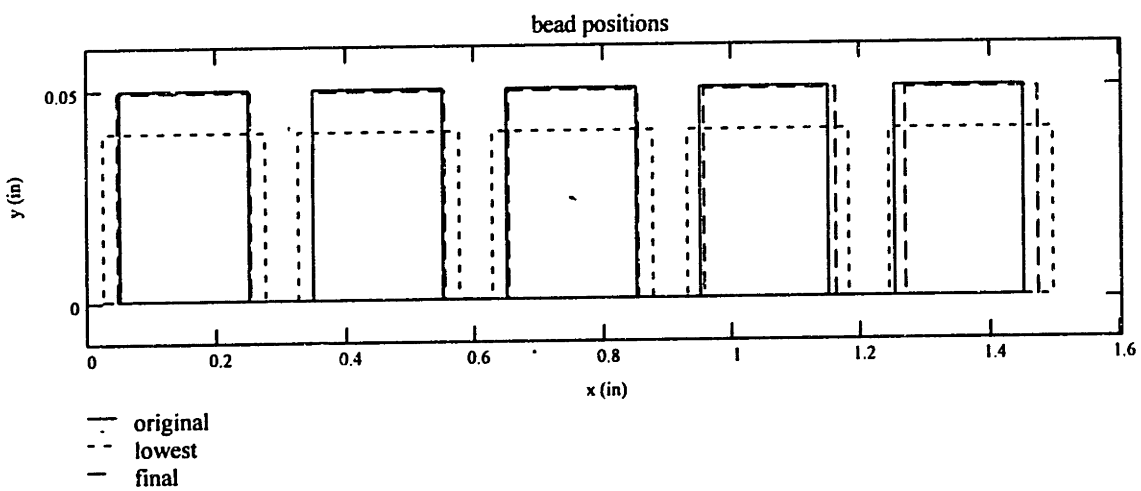


Figure C.10 Bead heights and widths at three different times.

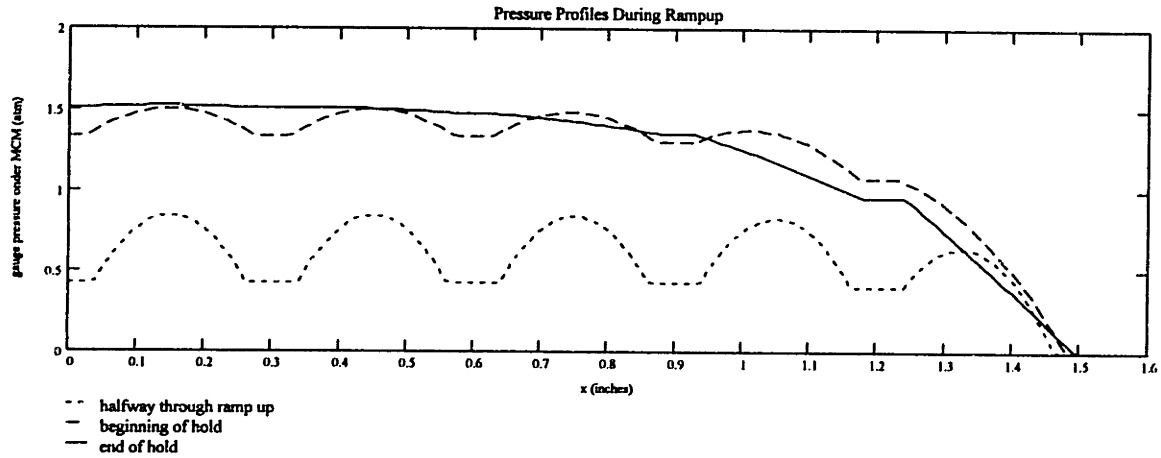


Figure C.11 Pressure profile changes during force ramp-up and hold.

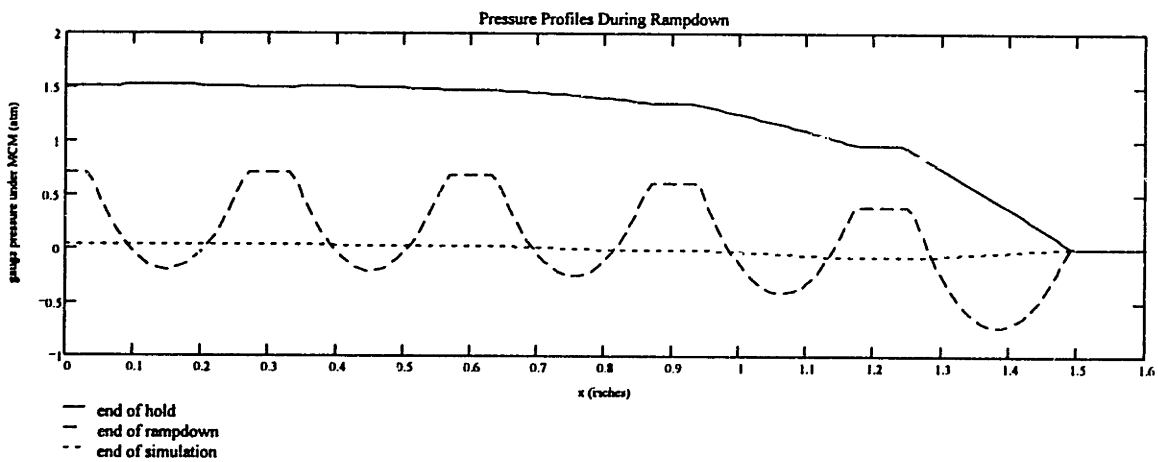


Figure C.12 Pressure profile changes during and after force ramp-down.

CONCLUSIONS

The purpose of this preliminary analysis has been to explore the basic physics of the adhesive flow problem in second level electronics packaging. Many of the assumptions used are tentative and the numerical values plugged into the expressions have little physical basis. It is hoped that this document will merely suggest what kind of answers we may be able to get from more complete modeling. It appears that, for solid patterns of adhesive, reasonably accurate closed form solutions can be attained. Also, for patterns with entrapped air pockets, numerical solutions can be provided that may capture some of the most important features of the real system's behavior.

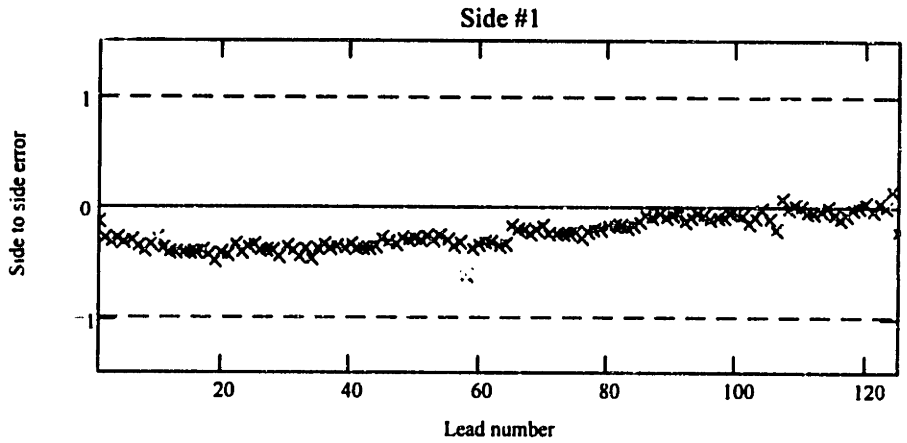
Some conjectures can be made on the basis of this the analysis. It appears that when the leads of an MCM fall short of the pads, that application of additional force or application of the same force over a longer time is not a promising solution. The low negative exponents in Equations 6 and 11 show that most of the downward motion will occur when the force is first applied with drastically diminishing returns after that.

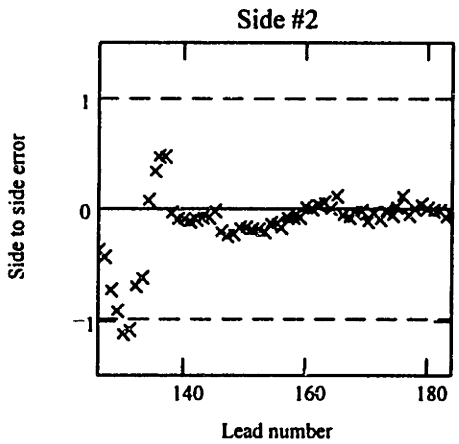
D. Data from MCM Surface Mount

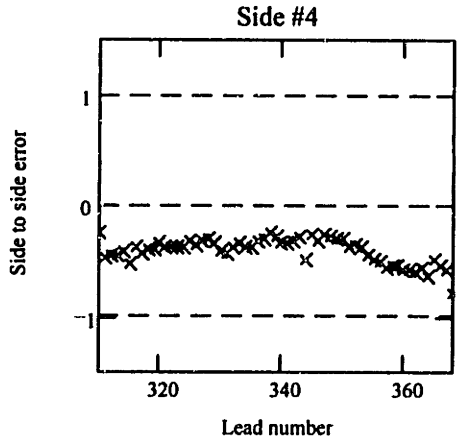
This Appendix presents data from experiments on the process of surface mount of multichip modules (MCMs). The experimental procedure and masking procedure is described in Section 2.9. For each of the nine MCMs manufactured and placed, there are four plots. Each plot displays the side-to-side error of each lead.

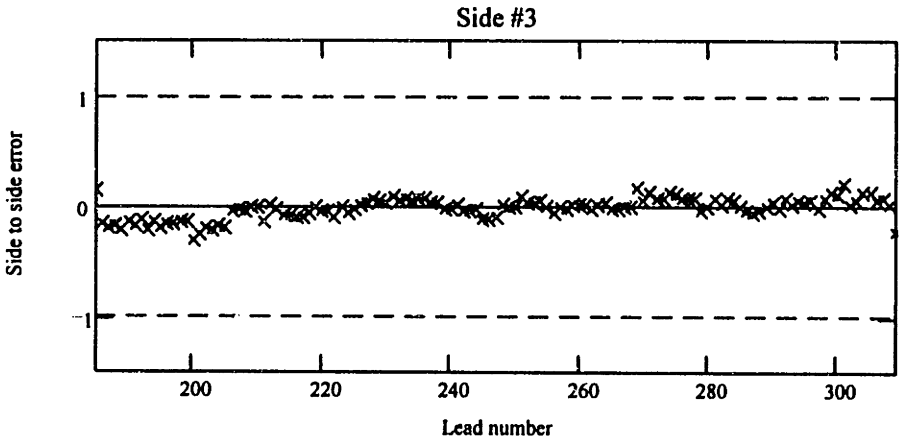
The data were normalized with respect to an arbitrary "tolerance width" on the lead positions in order to mask the capabilities of the actual experimental system. Therefore values in the plots are dimensionless. The dashed lines in the plots at 1 and -1 represent the arbitrary tolerance limits so that any data points outside the band represent non-conforming leads.

MCM #1 1/17/97

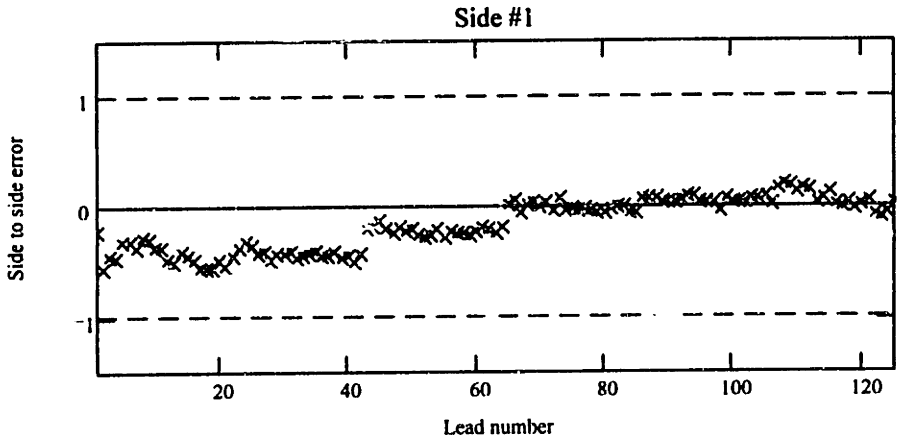

Side #1
Side to side error
1
0
-1
20
40
60
80
100
120
Lead number

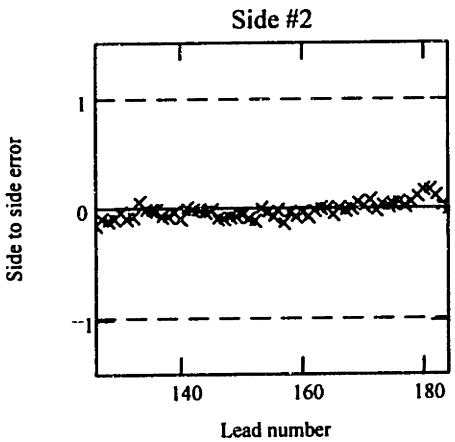

Side #2
Side to side error
1
0
-1
140
160
180
Lead number

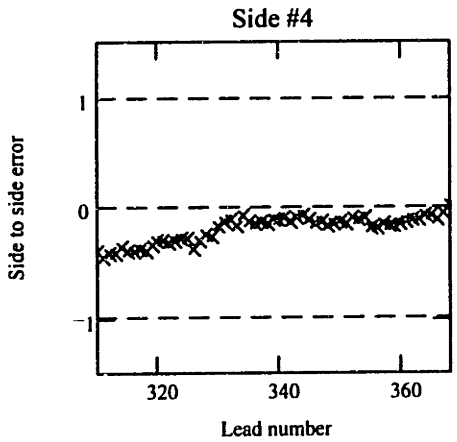

Side #4
Side to side error
1
0
-1
320
340
360
Lead number

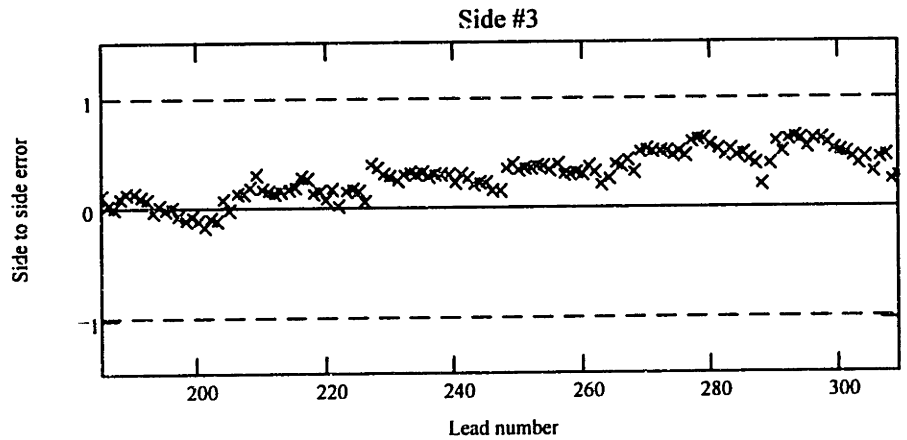

Side #3
Side to side error
1
0
-1
200
220
240
260
280
300
Lead number

MCM #2 1/17/97

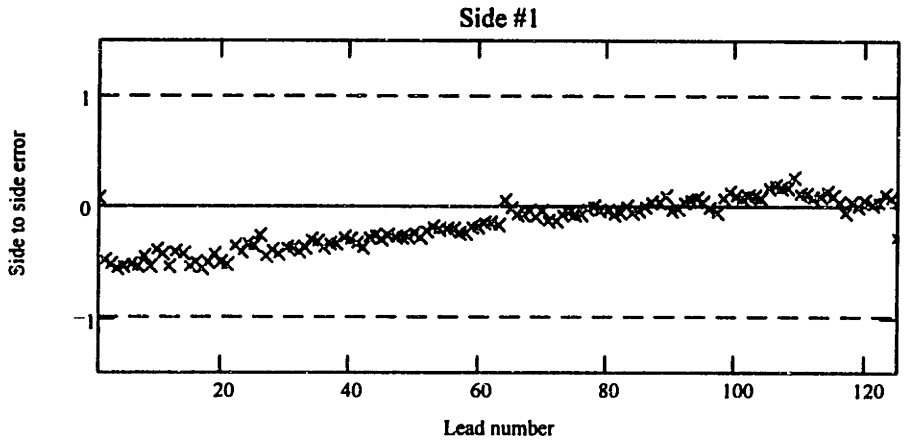

Side #1
Side to side error
1
0
-1
20
40
60
80
100
120
Lead number

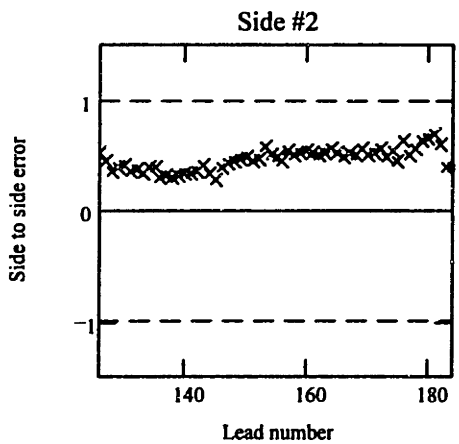

Side #2
Side to side error
1
0
-1
140
160
180
Lead number

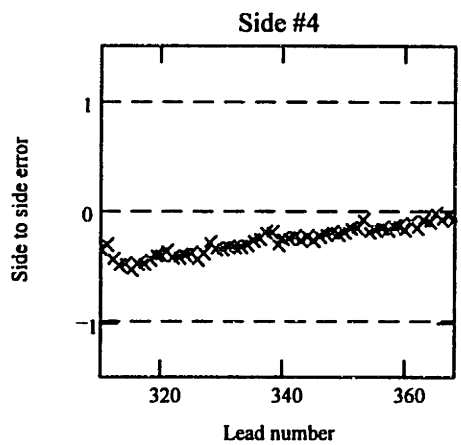

Side #4
Side to side error
1
0
-1
320
340
360
Lead number

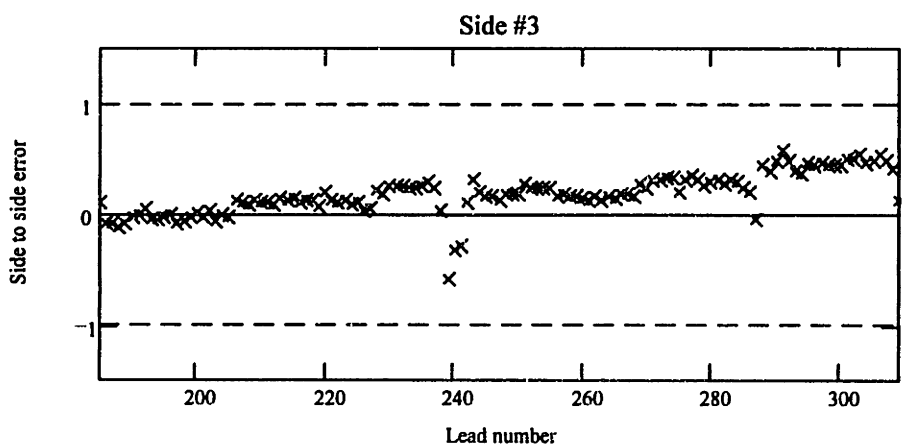

Side #3
Side to side error
1
0
-1
200
220
240
260
280
300
Lead number

MCM #3 1/17/97

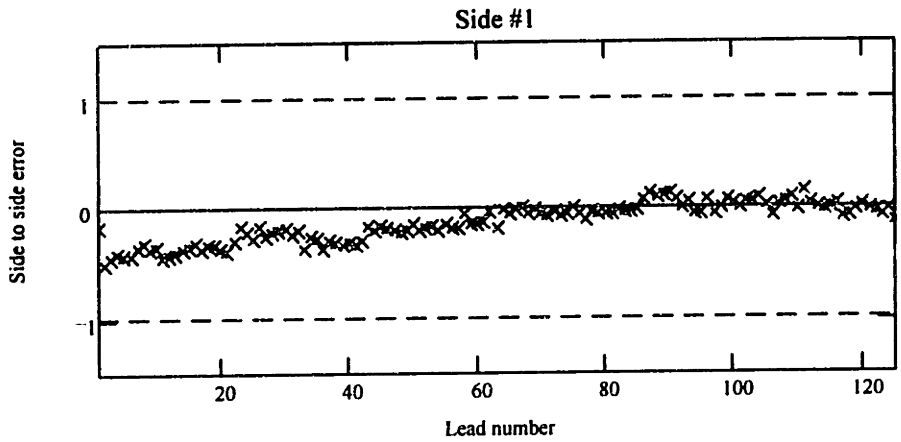

Side #1
Side to side error
1
0
-1
20
40
60
80
100
120
Lead number

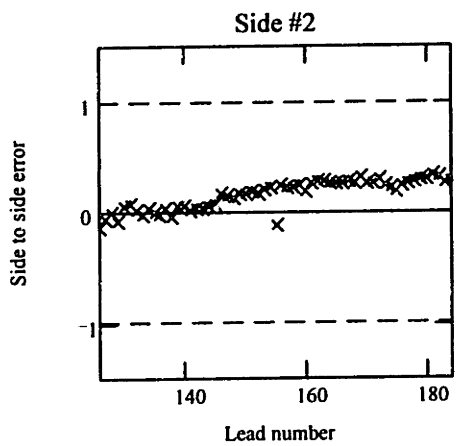

Side #2
Side to side error
1
0
-1
140
160
180
Lead number

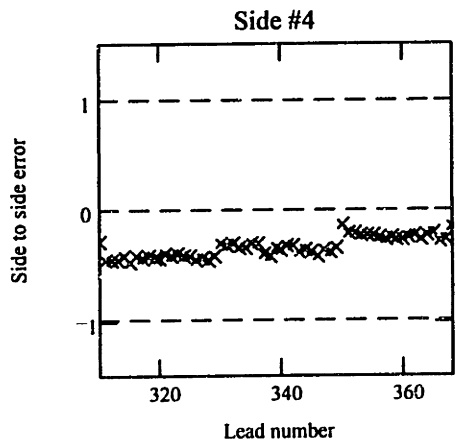

Side #4
Side to side error
1
0
-1
320
340
360
Lead number

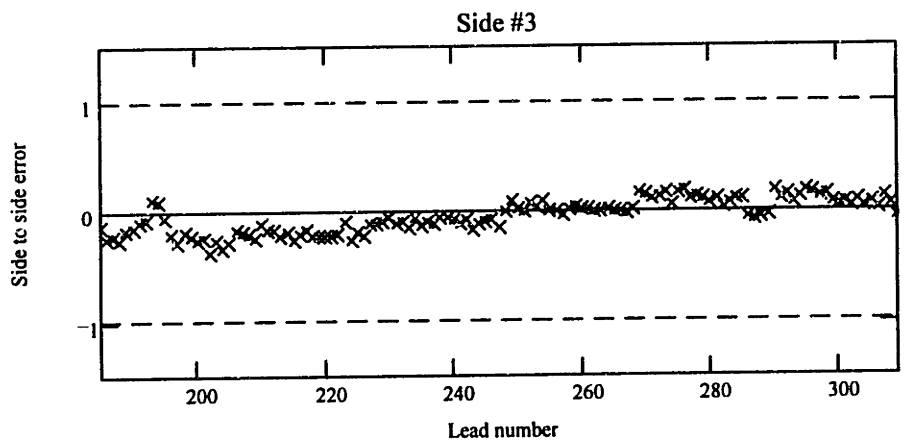

Side #3
Side to side error
1
0
-1
200
220
240
260
280
300
Lead number

MCM #4 1/17/97

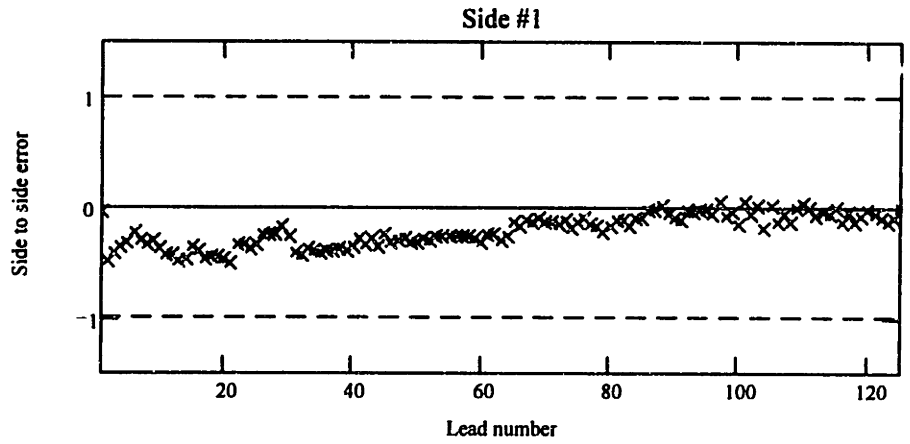

Side #1
Side to side error
1
0
-1
20
40
60
80
100
120
Lead number

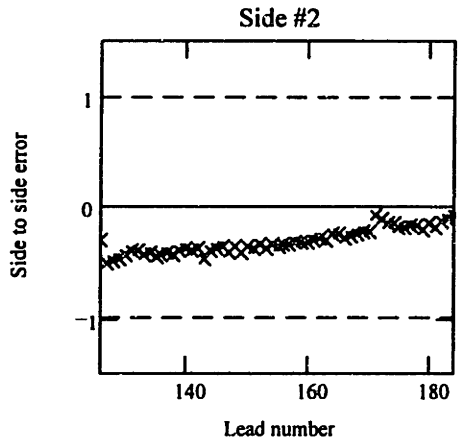

Side #2
Side to side error
1
0
-1
140
160
180
Lead number

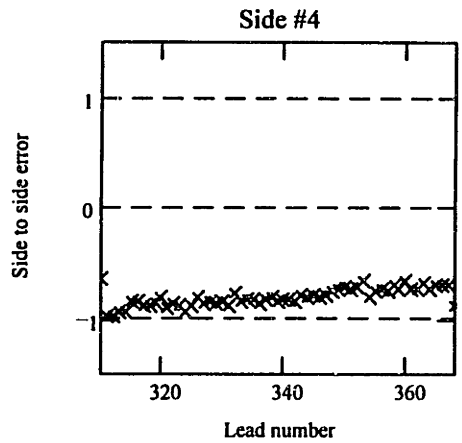

Side #4
Side to side error
1
0
-1
320
340
360
Lead number

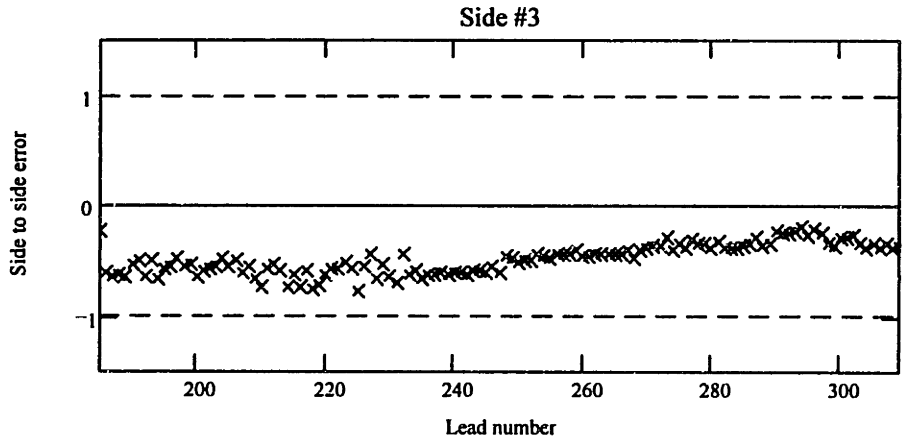

Side #3
Side to side error
1
0
-1
200
220
240
260
280
300
Lead number

MCM #5 1/17/97

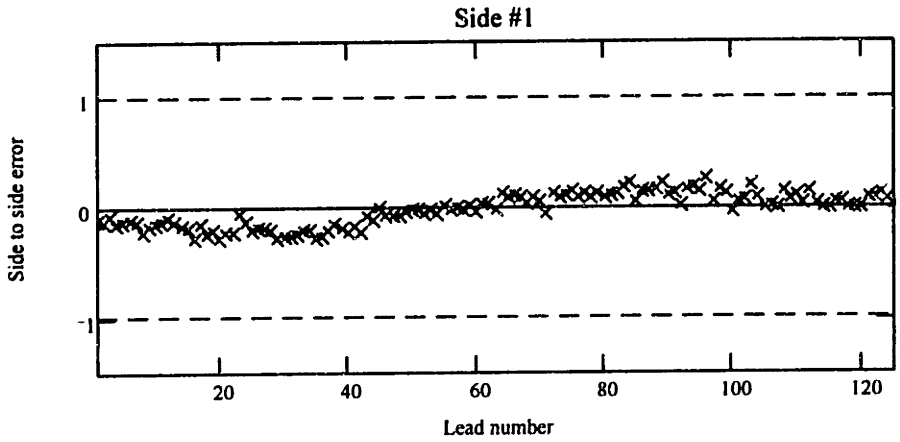

Side #1
Side to side error
1
0
-1
20
40
60
80
100
120
Lead number

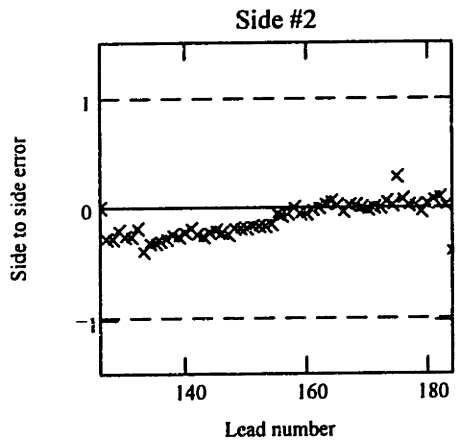

Side #2
Side to side error
1
0
-1
140
160
180
Lead number

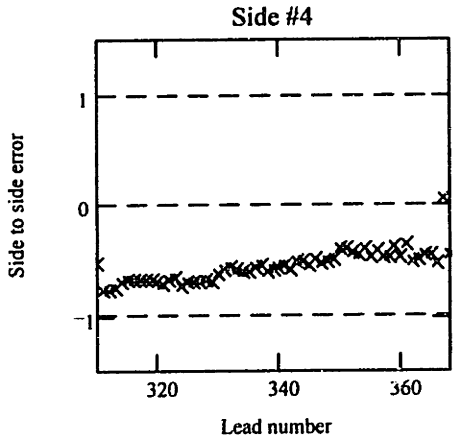

Side #4
Side to side error
1
0
-1
320
340
360
Lead number

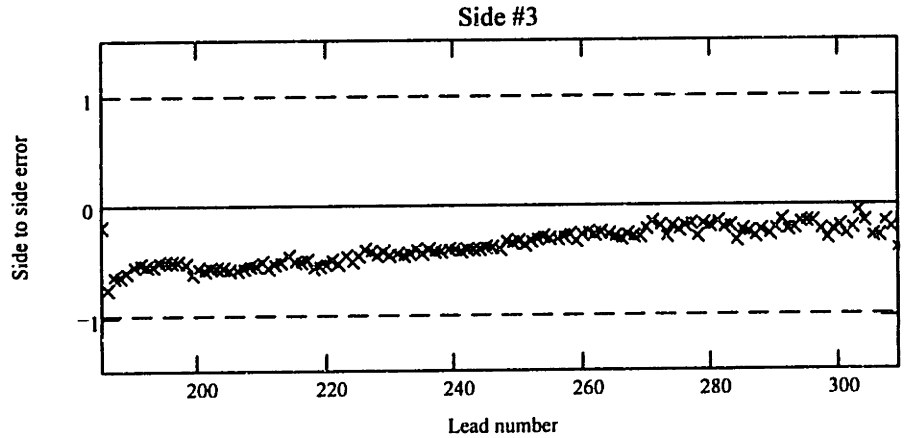

Side #3
Side to side error
1
0
-1
200
220
240
260
280
300
Lead number

MCM #6 1/17/97

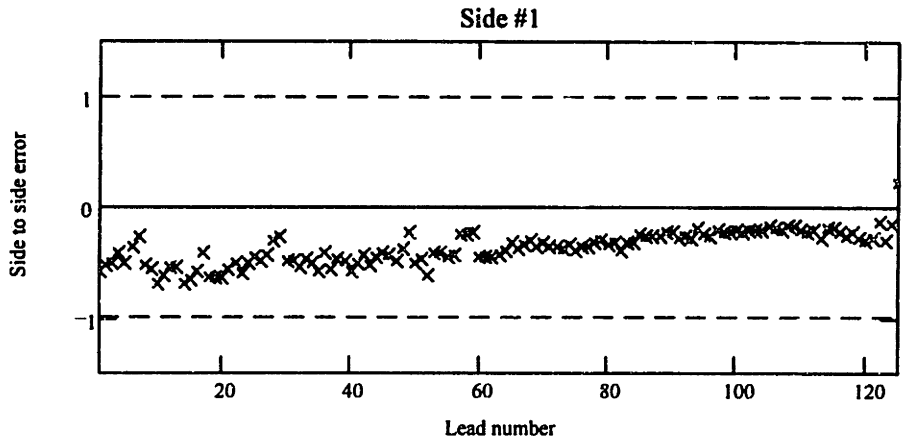

Side #1
Side to side error
1
0
-1
20
40
60
80
100
120
Lead number

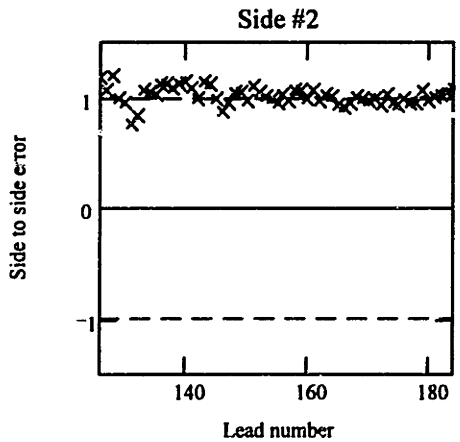

Side #2
Side to side error
1
0
-1
140
160
180
Lead number

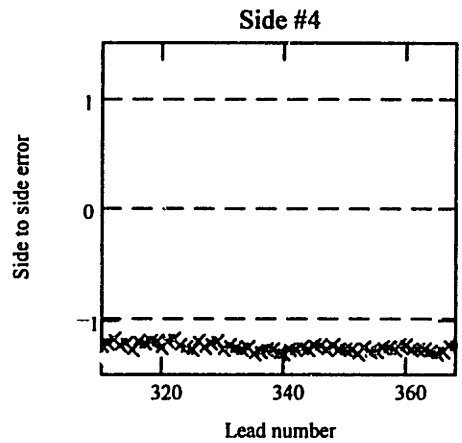

Side #4
Side to side error
1
0
-1
320
340
360
Lead number

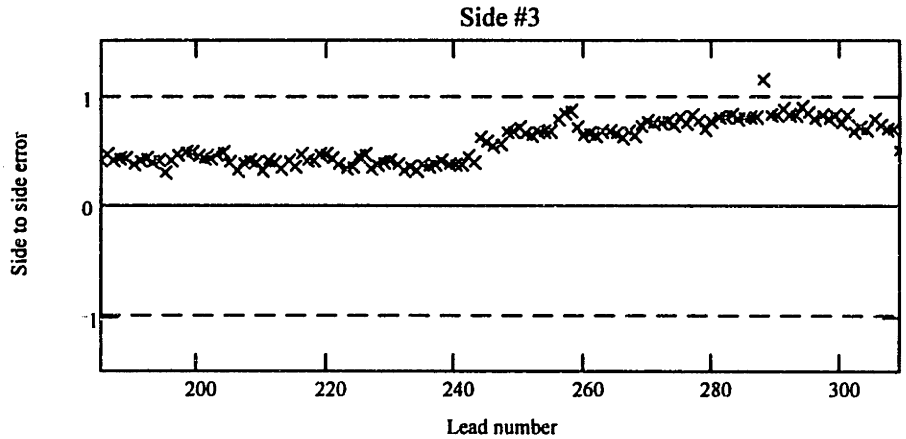

Side #3
Side to side error
1
0
-1
200
220
240
260
280
300
Lead number

MCM #7 1/17/97

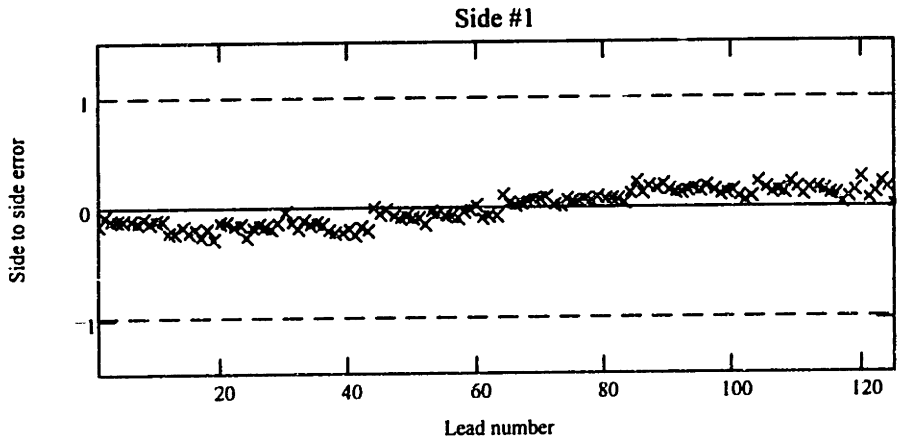

Side #1
Side to side error
1
0
-1
20
40
60
80
100
120
Lead number

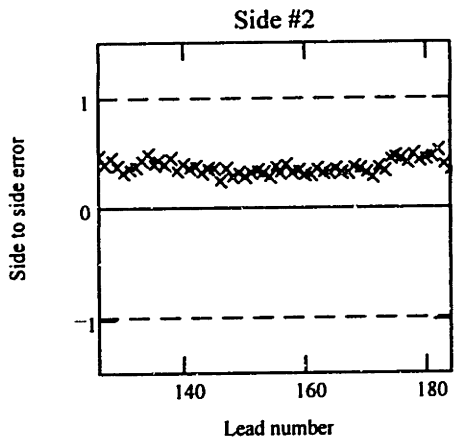

Side #2
Side to side error
1
0
-1
140
160
180
Lead number

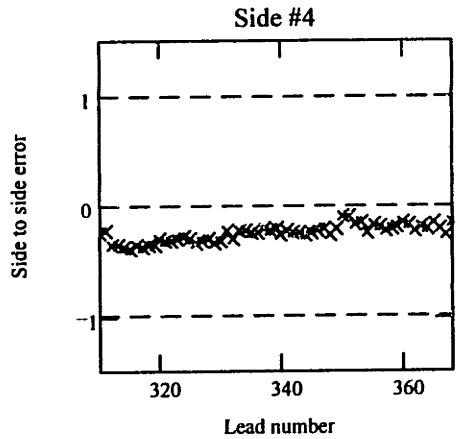

Side #4
Side to side error
1
0
-1
320
340
360
Lead number

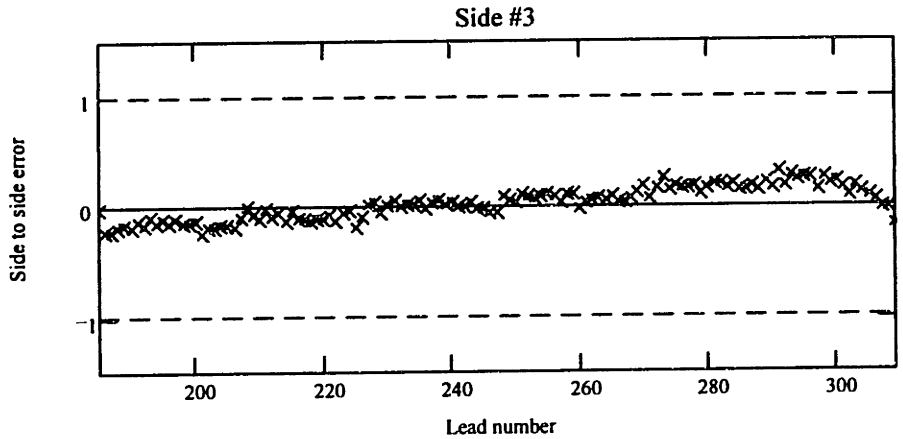

Side #3
Side to side error
1
0
-1
200
220
240
260
280
300
Lead number

MCM #8 1/17/97

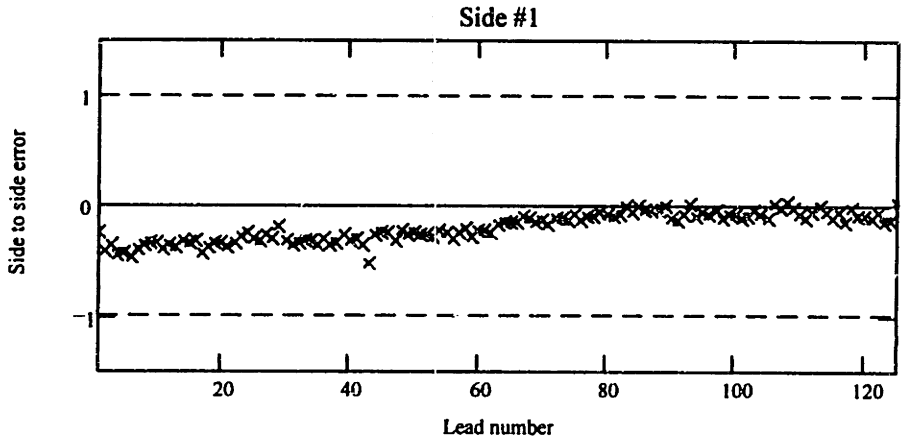

Side #1
Side to side error
1
0
-1
20
40
60
80
100
120
Lead number

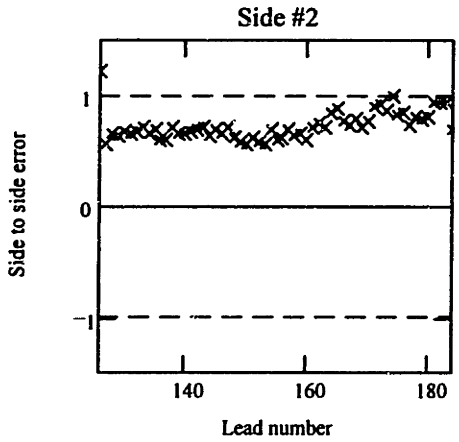

Side #2
Side to side error
1
0
-1
140
160
180
Lead number

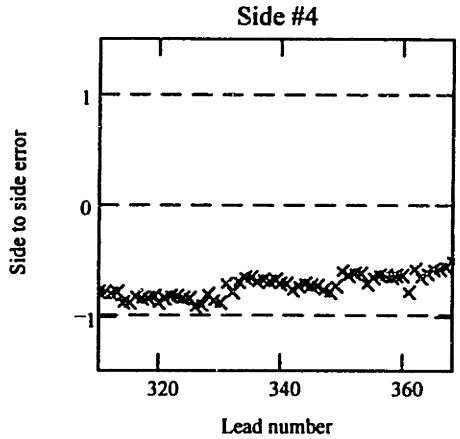

Side #4
Side to side error
1
0
-1
320
340
360
Lead number

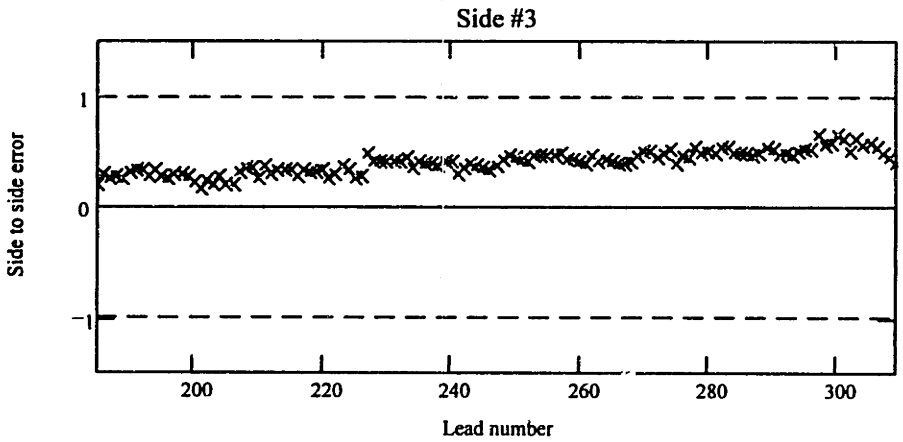

Side #3
Side to side error
1
0
-1
200
220
240
260
280
300
Lead number

MCM #9 1/17/97


Side #1
Side to side error
1
0
−1
20
40
60
80
100
120
Lead number


Side #2
Side to side error
1
0
−1
140
160
180
Lead number


Side #4
Side to side error
1
0
−1
320
340
360
Lead number


Side #3
Side to side error
1
0
−1
200
220
240
260
280
300
Lead number

E. Virtual Machining Model of CNC Crankpin Grinding

Virtual Machining Model for CNC Crankpin Grinding

Define constants to be used in the model

ORIGIN $:= 1$

I will use "1" as the origin of all matrices.

$\varepsilon := 10^{-6}$

TRUE $:= 1$

FALSE $:= 0$

Define some derived units that will be useful later. SI will be used in this document.

$\mu m := 10^{-6} \cdot m$

$\mu rad := 10^{-6} \cdot rad$

$mrad := 10^{-3} \cdot rad$

$rev := 2 \cdot \pi \cdot rad$

$rpm := \frac{rev}{min}$

Coefficient of thermal expansion of steel

$\alpha_{steel} := 10.8 \cdot 10^{-6} \cdot K^{-1}$

This value varies by about 20% depending upon the particular grade and heat treatment of the steel.

Young's modulus of steel

$E_{steel} := 207 \cdot 10^{6} \cdot Pa$

Define some assumptions used in the calculations

index_compensation_on := FALSE

If index compensation is on, the index error in the #1 pin, middle cut, is forced to zero. All other data is adjusted to reflect the change.

size_compensation_on := FALSE

If size compensation is on, then the size error in the #4 pin, middle cut is forced to zero. All the other pins are adjusted by the same amount.

Define the target values of the process

Nominal crankpin radius

$r_c(\theta) := 31.75mm$

Grinding wheel radius

$r_w := 304.8mm$

Nominal throw

$throw := 44.196mm$

Work spindle angle

This is the **independent** process variable under this control system.

$angular\_increment := 5 \cdot deg$

The size of each discrete step in the simulation in degrees.

Number of discrete steps in rotation of the crankshaft in degrees.

$steps := \frac{360 \cdot deg}{angular\_increment}$

$ang := 1 .. steps$

"ang" is an index range variable for arrays used extensively in the computations below.

$a_{ang} := (ang - 1) \cdot angular\_increment$

An array of evenly spaced angular steps.

$number\_of\_pins \equiv 6$

Number of crank pins on the crankshaft

$number\_of\_cuts \equiv 3$

$$cut\_z := \begin{pmatrix} -0.3 \\ 0 \\ 0.3 \end{pmatrix} \cdot in$$

The axial location of the "slices" through the crankpin for which a radial error plot is to be computed and displayed.

Define the location of the pins

Pin Z offset

The Z distance from the left side of the main bearing to the center of the pin.

$pin\_z\_offset_1 := 68.965mm$

$pin\_z\_offset_2 := 95.355mm$

$pin\_z\_offset_3 := 180.725mm$

$pin\_z\_offset_4 := 207.115mm$

$pin\_z\_offset_5 := 292.485mm$

$pin\_z\_offset_6 := 318.875mm$

Pin angular offset

The angular offset of the current pin from pin #1. Use the right hand rule for sign.

$pin\_ang\_offset_1 := 0 \cdot deg$

$pin\_ang\_offset_2 := 30 \cdot deg$

$pin\_ang\_offset_3 := 240 \cdot deg$

$pin\_ang\_offset_4 := 270 \cdot deg$

$pin\_ang\_offset_5 := 120 \cdot deg$

$pin\_ang\_offset_6 := 150 \cdot deg$

Distance carriage travels per rev of the lead screw.

$lead := 25 \cdot mm$

Rate of rotation of the crankshaft.

20-25 rpm

$\omega_{crankshaft} := 25 \cdot rpm$

The control function which reciprocates the x axis carriage to create the correct nominal geometry of the crankpins.

$$x(a) := \left(r_c(0) + r_w + throw\right) - \left[throw \cdot \cos(a) + \sqrt{\left(r_c(0) + r_w\right)^2 - (throw \cdot \sin(a))^2}\right]$$

Define the error sources

X Carriage

Z_Carriage

$$roll\_x(a) := \frac{0 \cdot \mu rad}{1 \cdot mm} \cdot x(a)$$

$$roll\_z(z) := \frac{0 \cdot \mu rad}{1 \cdot mm} \cdot z$$

These can change with Z position

(pin number)
These
can change
with X
position

$$pitch\_z(z) := \frac{0 \cdot \mu rad}{1 \cdot mm} \cdot z$$

$$pitch\_x(a) := \frac{0 \cdot \mu rad}{1 \cdot mm} \cdot x(a)$$

$$yaw\_z(z) := \frac{0 \cdot \mu rad}{1 \cdot mm} \cdot z$$

$$yaw\_x(a) := \frac{0 \cdot \mu rad}{1 \cdot mm} \cdot (x(a) - throw)$$

$$y\_str\_x(a) := \frac{0 \cdot \mu m}{1 \cdot mm} \cdot (x(a) - throw)$$

$$x\_str\_z(z) := \frac{0 \cdot \mu m}{1 \cdot mm} \cdot z$$

If these straightness
errors are def as
constant, then
they are offsets.
z_str_x non-sensitive

$$y\_str\_z(z) := \frac{0 \cdot \mu m}{1 \cdot mm} \cdot z$$

$y\_sqr\_x := 0 \cdot \mu rad$
100μrad max
$x\_sqr\_z := 0 \cdot deg$
$z\_sqr\_x := 0 \cdot deg$
$y\_sqr\_z := 0 \cdot deg$
$x\_par\_tool\_spindle := 0 \cdot deg$
$x\_par\_work\_spindle := 0 \cdot \mu rad$
10μrad max
$y\_par\_tool\_spindle := 0 \cdot deg$
$y\_par\_work\_spindle := 0 \cdot mrad$
1 mrad max
Grinding wheel wear not compensated for in
the control algorithm measured radially.
Can be as much as 3mm (the whole thickness of the CBN coating).
$wheel\_wear := 0 \cdot \mu m$
Actual radius of the grinding wheel after wear.
$rw_{act} := r_w - wheel\_wear$

$$cum\_lead := 1 \cdot \frac{\mu m}{mm}$$

Cumulative screw lead error (15μm/300mm)
Once per revolution lead error (5-10 μm).

$$per\_rev\_lead := 0 \cdot \frac{\mu m}{rev}$$

Lost Motion (similar to backlash, but for
preloaded bearings). X axis only.

$lost\_motion := 0 \cdot \mu m$

$$lost\_motion(a) := \begin{cases} \frac{lost\_motion}{2} & \text{if } (x(a+\varepsilon) - x(a)) > 0 \cdot mm \\ \frac{-lost\_motion}{2} & \text{otherwise} \end{cases}$$

Error in diameter of the main bearing (up to 25μm)

$main\_bearing\_size\_err := 0 \cdot \mu m$

Due to the geometry of the fixture, error in size of the main bearing will cause a Y offset of the main bearing centerline from the work spindle centerline.

Y offset of main bearing from the spindle axis at TDC

$$main\_err_x := \frac{\sqrt{2}}{2} \cdot main\_bearing\_size\_err$$

Pitch of main bearing relative to the spindle

$pitch\_w := 0 \cdot \mu rad$

put in terms of size mismatch of left and right main bearings

(crankshaft misalignment in the fixture)

Yaw error of main bearing relative to the spindle

$yaw\_w := 0 \cdot \mu rad$

(crankshaft misalignment in the fixture)

42μrad if #1 undersized and #2 oversized 12.5μm

(add -8.7 main_err.x to rotate around #4 pin).

Force between the grinding wheel and work

(about 25 pounds).

$$F_{grind}(a) := 0 \cdot \left(\cos\left(\operatorname{asin}\left(\frac{\sin(a) \cdot throw}{r_w + r_c(a)}\right)\right)\right) \cdot lbf$$

System stiffness of the grinder side

$$K_{sys} := 700000 \frac{lbf}{in}$$

Stiffness of the grinding wheel spindle.

$$K_{spindle} := 10^6 \cdot \frac{lbf}{in}$$

$$K_{rest} := \frac{1}{K_{sys}^{-1} - K_{spindle}^{-1}}$$

$$\frac{K_{rest}}{\left(\frac{lbf}{in}\right)} = 2.333 \cdot 10^6$$

Stiffness of the rest of the system.

Mass of the X axis carriage (or, more properly, the weight not already compensated for)

$x\_carriage\_mass := 0 \cdot lb$

Additional force on the lead screw due to this acceleration

$$F_{acc}(a) := 0 \cdot x\_carriage\_mass \cdot \omega_{crankshaft}^2 \cdot m$$

Define the HTM functions

These will serve as a shorthand notation for calculation of the HTM model of the machine making the section in which you fill the matrices much neater. Because of these function definitions, you don't need to repeat the ones and zeros below over and over for each carriage.

Rotation about the x axis
Rotation about the y axis

$$\mathbf{R_x}(\varepsilon_x) := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\varepsilon_x) & -\sin(\varepsilon_x) & 0 \\ 0 & \sin(\varepsilon_x) & \cos(\varepsilon_x) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{R_y}(\varepsilon_y) := \begin{bmatrix} \cos(\varepsilon_y) & 0 & \sin(\varepsilon_y) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\varepsilon_y) & 0 & \cos(\varepsilon_y) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotation about the z axis
Translation in all 3 directions

$$\mathbf{R_z}(\varepsilon_z) := \begin{bmatrix} \cos(\varepsilon_z) & -\sin(\varepsilon_z) & 0 & 0 \\ \sin(\varepsilon_z) & \cos(\varepsilon_z) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}(\delta_x, \delta_y, \delta_z) := \begin{bmatrix} 1 & 0 & 0 & \delta_x \cdot m^{-1} \\ 0 & 1 & 0 & \delta_y \cdot m^{-1} \\ 0 & 0 & 1 & \delta_z \cdot m^{-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

I convert the δ's to dimensionless quantities
because MathCad doesn't like matrices w/ mixed units.
The HTM function takes 6 parameters that describe a rigid body motion and
assembles them into and returns a 4X4 homogeneous transformation matrix.
An HTM can be premultiplied by a vector to give it a rigid body motion.
Note that the order of rotations is important. If you want to perform rotations
about more than one axis in a single HTM, you should be careful.
HTM performs rotations first, then translations.

$$\mathbf{HTM}(d,a) := \mathbf{T}(d_1, d_2, d_3) \cdot \mathbf{R_x}(a_1) \cdot \mathbf{R_y}(a_2) \cdot \mathbf{R_z}(a_3)$$

The function linHTM is a linearized version of HTM. It works if the angles (ε) are small.

$$\mathbf{linHTM}(d,a) := \begin{bmatrix} 1 & -a_3 & a_2 & d_1 \cdot m^{-1} \\ a_3 & 1 & -a_1 & d_2 \cdot m^{-1} \\ -a_2 & a_1 & 1 & d_3 \cdot m^{-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Define the HTM matrices for each rigid body in the machine.

Base to Work

For each body, send the appropriate values to the HTM function.
Start at the base and work your way out to the work.
For the CNC crankpin grinder, the center of stiffness of the right side work spindle at the rightmost travel of the Z axis carriage is considered the origin of a reference coordinate system.

Z Carriage (W1)

X distance of Z axis lead screw from work spindle centerline
Ydistance of Z axis lead screw from work spindle centerline
Z distance of Z axis lead screw from the bed centerline (there is no well defined z-axis ctr of stiffness).
Home (W1)

$$H_{W1} := HTM\left[\begin{pmatrix} -120\cdot mm + 794\cdot mm\cdot \sin(x\_sqr\_z) \\ -550\cdot mm - 794\cdot \sin(y\_sqr\_z)\cdot mm \\ 794\cdot mm \end{pmatrix}, \begin{pmatrix} 0\cdot deg + y\_sqr\_z \\ 0\cdot deg + x\_sqr\_z \\ 0\cdot deg \end{pmatrix}\right]$$

Distance to move the Z carriage from the left end of the working range to align the grinding wheel with the pin.
Pose (W1)

$$P_{W1}(p,z) := HTM\left[\begin{bmatrix} 0\cdot mm \\ 0\cdot mm \\ -pin\_z\_offset_p + z \end{bmatrix}, \begin{pmatrix} 0\cdot deg \\ 0\cdot deg \\ 0\cdot deg \end{pmatrix}\right]$$

For now, error motions of the Z axis carriage are taken to be zero.
Error (W1)

$$E_{W1}(p) := linHTM\left[\begin{bmatrix} x\_str\_z(pin\_z\_offset_p) \\ y\_str\_z(pin\_z\_offset_p) \\ 0\cdot \mu m \end{bmatrix}, \begin{bmatrix} pitch\_z(pin\_z\_offset_p) \\ yaw\_z(pin\_z\_offset_p) \\ roll\_z(pin\_z\_offset_p) \end{bmatrix}\right]$$

Work Spindle (W2)
This vector brings the coordinate system back from the Z axis carriage to the .
Home (W2)

$$H_{W2} := HTM\left[\begin{pmatrix} 120\cdot mm \\ 550\cdot mm \\ -794\cdot mm \end{pmatrix}, \begin{pmatrix} 0\cdot deg \\ 0\cdot deg \\ 0\cdot deg \end{pmatrix}\right]$$

The work spindle rotates about the Z axis w.r.t. the Z axis carriage.
The angle of rotation "a" varies from 0 to 2π.
Pin angular offset is subtracted so that the pin to be ground will pointing in the positive x direction from the main bearing when "a" equals zero. If not, the control equation would be grossly in error.
Pose (W2)

$$P_{W2}(a,p) := HTM\left[\begin{pmatrix} 0\cdot mm \\ 0\cdot mm \\ 0\cdot mm \end{pmatrix}, \begin{bmatrix} 0\cdot deg \\ 0\cdot deg \\ a + pin\_ang\_offset_p \end{bmatrix}\right]$$

For now, disregard errors in the work spindle.
Error (W2)

$$E_{W2} := linHTM\left[\begin{bmatrix} 0\cdot \mu m \\ 0\cdot \mu m \\ 0\cdot \mu m \end{bmatrix}, \begin{pmatrix} y\_par\_work\_spindle \\ x\_par\_work\_spindle \\ 0\cdot deg \end{pmatrix}\right]$$

Main Bearings (W3)
The Z offset of the left face of the crankshaft main bearings from the work spindle center of stiffness.
Home (W3)

$$H_{W3} := HTM\left[\begin{pmatrix} 0\cdot mm \\ 0\cdot mm \\ 0\cdot mm \end{pmatrix}, \begin{pmatrix} 0\cdot deg \\ 0\cdot deg \\ 0\cdot deg \end{pmatrix}\right]$$

The main bearing should be stationary with respect to the work spindle.
Pose (W3)

$$P_{W3}(p) := HTM\left[\begin{pmatrix}0\cdot mm\\0\cdot mm\\0\cdot mm\end{pmatrix},\begin{bmatrix}0\cdot deg\\0\cdot deg\\-pin\_ang\_offset_p\end{bmatrix}\right]$$

The error in work fixturing causes an offset of the main bearing centerline from the work spindle centerline.

Error (W3)

$$E_{W3} := linHTM\left[\begin{bmatrix}main\_err_x\\0\cdot\mu m\\0\cdot\mu m\end{bmatrix},\begin{pmatrix}pitch\_w\\yaw\_w\\0\cdot deg\end{pmatrix}\right]$$

Crankpin (W4)

The crankpin is offset from the main bearing centerline by half the stroke length (throw). For each pin, the +X direction goes from the main bearing centerline to the pin centerline. Each pin is also offset from the main bearing by some axial distance. Each pin's angular offset is measured from the #1 pin.

Home (W4)

$$H_{W4}(p) := HTM\left[\begin{bmatrix}throw\\0\cdot mm\\pin\_z\_offset_p\end{bmatrix},\begin{pmatrix}0\cdot deg\\0\cdot deg\\0\cdot deg\end{pmatrix}\right]$$

The pose matrix is zero because the crankpin is fixed to the crankshaft main bearing.

Pose (W4)

$$P_{W4} := HTM\left[\begin{pmatrix}0\cdot mm\\0\cdot mm\\0\cdot mm\end{pmatrix},\begin{pmatrix}0\cdot deg\\0\cdot deg\\0\cdot deg\end{pmatrix}\right]$$

Error (W4)

$$E_{W4} := linHTM\left[\begin{bmatrix}0\cdot\mu m\\0\cdot\mu m\\0\cdot\mu m\end{bmatrix},\begin{pmatrix}0\cdot deg\\0\cdot deg\\0\cdot deg\end{pmatrix}\right]$$

Base to Tool

For each body, send the appropriate values to the HTM function.

Start at the base and work your way out to the tool.

X Carriage (T1)

X distance of work spindle to X carriage center of stiffness.

Y distance of work spindle to X carriage lead screw.

Z distance of left side work spindle center of stiffness to X carriage lead screw when the Z carriage is at its home position per Landis drawings.

Home (T1)

$$H_{T1}(a) := HTM\left[\begin{bmatrix}r_c(a)+r_w+throw-28.32in+z\_sqr\_x\cdot(390.22mm+304.8mm)\\-7.13in\\390.22mm+304.8mm\end{bmatrix},\begin{pmatrix}0\cdot deg\\z\_sqr\_x\\0\cdot deg\end{pmatrix}\right]$$

The correction for wheel wear is made to correct for size (but not corrected in the control algorithm!).

Pose (T1)

$$P_{T1}(a) := HTM\left[\begin{pmatrix}-x(a)-wheel\_wear\\0\cdot mm\\0\cdot mm\end{pmatrix},\begin{pmatrix}0\cdot deg\\0\cdot deg\\y\_sqr\_x\end{pmatrix}\right]$$

Error (T1)

$$\varepsilon_{xscrew}(a) := -(x(a)\cdot cum\_lead) + \frac{per\_rev\_lead}{2}\cdot\sin\left(2\cdot\pi\cdot\frac{x(a)}{lead}\right) + \frac{F_{grind}(a)}{K_{sys}} + \frac{F_{acc}(a)}{K_{rest}} + lost\_motion(a)$$

$$E_{T1}(a) := linHTM\left[\begin{bmatrix}\varepsilon_{xscrew}(a)\\ y\_str\_x(a)\\ 0\cdot\mu m\end{bmatrix}, \begin{pmatrix}roll\_x(a)\\ yaw\_x(a)\\ pitch\_x(a)\end{pmatrix}\right]$$

Grinder Spindle (T2)

X distance of X carriage center of stiffness to grinder spindle center of stiffness.
Y distance of X carriage lead screw to grinder spindle center of stiffness..
Z distance of X carriage lead screw when the Z carriage is at its home position per Landis drawings to grinder spindle center of stiffness..

Home (T2)

$$H_{T2} := HTM\left[\begin{pmatrix}28.32\,in\\ 7.13\,in\\ 0\cdot mm\end{pmatrix}, \begin{pmatrix}0\cdot deg\\ 0\cdot deg\\ 0\cdot deg\end{pmatrix}\right]$$

Pose (T2)

$$P_{T2} := HTM\left[\begin{pmatrix}0\cdot mm\\ 0\cdot mm\\ 0\cdot mm\end{pmatrix}, \begin{pmatrix}0\cdot deg\\ 0\cdot deg\\ 0\cdot deg\end{pmatrix}\right]$$

Error (T2)

$$E_{T2} := linHTM\left[\begin{bmatrix}0\cdot\mu m\\ 0\cdot\mu m\\ 0\cdot\mu m\end{bmatrix}, \begin{pmatrix}0\cdot deg\\ 0\cdot deg\\ 0\cdot deg\end{pmatrix}\right]$$

Grinding Wheel Center (T3)

X distance should be zero.
Y distance should be zero.
Z distance from the grinder spindle center of stiffness to the grinding wheel center.

Home (T3)

$$H_{T3} := HTM\left[\begin{bmatrix}0\cdot mm\\ 0\cdot mm\\ -(230\cdot mm + 74.6\,mm + 23\cdot mm + 367.42\,mm)\end{bmatrix}, \begin{pmatrix}x\_par\_tool\_spindle\\ y\_par\_tool\_spindle\\ 0\cdot deg\end{pmatrix}\right]$$

Pose (T3)

$$P_{T3} := HTM\left[\begin{pmatrix}0\cdot mm\\ 0\cdot mm\\ 0\cdot mm\end{pmatrix}, \begin{pmatrix}0\cdot deg\\ 0\cdot deg\\ 0\cdot deg\end{pmatrix}\right]$$

Error (T3)

$$E_{T3} := linHTM\left[\begin{bmatrix}0\cdot\mu m\\ 0\cdot\mu m\\ 0\cdot\mu m\end{bmatrix}, \begin{pmatrix}0\cdot deg\\ 0\cdot deg\\ 0\cdot deg\end{pmatrix}\right]$$

Apply the HTM matrices

This matrix takes coordinates on the tool into the base coordinate system.

$$B_T(a) := H_{T1}(a)\cdot E_{T1}(a)\cdot P_{T1}(a)\cdot H_{T2}\cdot E_{T2}\cdot P_{T2}\cdot H_{T3}\cdot E_{T3}\cdot P_{T3}$$

This matrix takes coordinates on the work into base coordinate system

$$B_W(a,p,z) := H_{W1}\cdot E_{W1}(p)\cdot P_{W1}(p,z)\cdot H_{W2}\cdot E_{W2}\cdot P_{W2}(a,p)\cdot H_{W3}\cdot E_{W3}\cdot P_{W3}(p)\cdot H_{W4}(p)\cdot E_{W4}\cdot P_{W4}$$

This section simulates the dressing operation

$$fit := \left| \begin{array}{l} \text{for } i \in 1..9 \\ \quad \left| \begin{array}{l} z \leftarrow \frac{i-5}{10} \cdot in \\ v \leftarrow B_T(0)^{-1} \cdot \left(H_{W1} \cdot E_{W1}(1) \cdot P_{W1}(1,z)\right) \cdot \begin{bmatrix} 0.500746 - .3048 \\ 0.550 \\ -0.725035 \\ 1 \end{bmatrix} \\ zv_i \leftarrow v_3 \\ rw_i \leftarrow \sqrt{(v_1)^2 + (v_2)^2} \end{array} \right. \\ fit \leftarrow augment(zv, rw) \\ fit \end{array} \right.$$

$$vs := cspline\left(fit^{<1>}, fit^{<2>}\right)$$

$$v_w(z) := interp\left(vs, fit^{<1>}, fit^{<2>}, z\right) \cdot m$$

Note that the vector has dimensions (length) while the HTM matrices are dimensionless.
The vector at the right of the equation above is the zero vector in this space. The one at the bottom is a dummy which makes the HTM calculations possible. Normally I would put in a vector representing the toolpoint offset from the final coordinate system, but on this machine the "toolpoint" is a complicated function of "a". For now, I just find the center of the grinding wheel. The math to find the "toolpoint" is worked out below.
Up to this point the sheet mostly defines functions. Now the results are actually computed and placed into arrays so that the can be used to process the information efficiently.

$$\underline{\text{tool\_to\_work}}(pin, angle, point) := submatrix\left[\left[\left(B_W(angle, pin, 0 \cdot m)\right)^{-1} \cdot B_T(angle) \cdot point\right] \cdot m^{-1}, 1, 3, 1, 1\right]$$

The argument pin is an index.
angle is an angle
point is a column vector with dimensions of length.

$$\nabla(v, p, a, c) := \frac{v(p, a + \varepsilon, c) - v(p, a - \varepsilon, c)}{\varepsilon}$$

A central difference gradient of a vector.

$$
\begin{aligned}
\underline{\mathbf{cut\_pt}} := \;& \text{for } pin \in 1..number\_of\_pins \\
& \quad \text{for } ang \in 1..steps \\
& \quad\quad \underline{\mathbf{R}} \leftarrow \underline{\mathbf{tool\_to\_work}}\left(pin, a_{ang}, (0\ \ 0\ \ 0\ \ 1)^T \cdot m\right) \\
& \quad\quad \underline{\mathbf{R}}_{\underline{\mathbf{dot}}} \leftarrow \nabla\left(\underline{\mathbf{tool\_to\_work}}, pin, a_{ang}, (0\ \ 0\ \ 0\ \ 1)^T \cdot m\right) \\
& \quad\quad \underline{\mathbf{A}} \leftarrow \underline{\mathbf{tool\_to\_work}}\left(pin, a_{ang}, (0\ \ 0\ \ 1\ \ 0)^T \cdot m\right) \\
& \quad\quad \underline{\mathbf{A}}_{\underline{\mathbf{dot}}} \leftarrow \nabla\left(\underline{\mathbf{tool\_to\_work}}, pin, a_{ang}, (0\ \ 0\ \ 1\ \ 0)^T \cdot m\right) \\
& \quad\quad \underline{\mathbf{e}}_{\underline{1}} \leftarrow \underline{\mathbf{A}} \\
& \quad\quad \underline{\mathbf{e}}_{\underline{3}} \leftarrow \frac{\underline{\mathbf{A}} \times \underline{\mathbf{R}}_{\underline{\mathbf{dot}}}}{\left|\underline{\mathbf{A}} \times \underline{\mathbf{R}}_{\underline{\mathbf{dot}}}\right|} \\
& \quad\quad \underline{\mathbf{e}}_{\underline{2}} \leftarrow \underline{\mathbf{e}}_{\underline{3}} \times \underline{\mathbf{e}}_{\underline{1}} \\
& \quad\quad \text{for } cut \in 1..number\_of\_cuts \\
& \quad\quad\quad z \leftarrow cut\_z_{cut} \\
& \quad\quad\quad a \leftarrow \underline{\mathbf{R}}_{\underline{\mathbf{dot}}} \underline{\mathbf{e}}_{\underline{2}} + r_w(z) \cdot m^{-1} \cdot \left(\frac{d}{dz} r_w(z)\right) \cdot \underline{\mathbf{A}}_{\underline{\mathbf{dot}}} \underline{\mathbf{e}}_{\underline{2}} + z \cdot m^{-1} \cdot \underline{\mathbf{A}}_{\underline{\mathbf{dot}}} \underline{\mathbf{e}}_{\underline{2}} \\
& \quad\quad\quad b \leftarrow \underline{\mathbf{R}}_{\underline{\mathbf{dot}}} \underline{\mathbf{e}}_{\underline{3}} + r_w(z) \cdot m^{-1} \cdot \left(\frac{d}{dz} r_w(z)\right) \cdot \underline{\mathbf{A}}_{\underline{\mathbf{dot}}} \underline{\mathbf{e}}_{\underline{3}} + z \cdot m^{-1} \cdot \underline{\mathbf{A}}_{\underline{\mathbf{dot}}} \underline{\mathbf{e}}_{\underline{3}} \\
& \quad\quad\quad c \leftarrow -\left(\frac{d}{dz} r_w(z)\right) \cdot \underline{\mathbf{R}}_{\underline{\mathbf{dot}}} \underline{\mathbf{e}}_{\underline{1}} \\
& \quad\quad\quad \cos\theta \leftarrow \frac{-\left(a \cdot c - c^2 + b^2 - b \cdot \sqrt{b^2 + a^2 - c^2}\right)}{\left(a^2 - a \cdot c + b^2 - b \cdot \sqrt{b^2 + a^2 - c^2}\right)} \\
& \quad\quad\quad \sin\theta \leftarrow (a - c) \cdot \frac{\left(b - \sqrt{b^2 + a^2 - c^2}\right)}{\left(a^2 - a \cdot c + b^2 - b \cdot \sqrt{b^2 + a^2 - c^2}\right)} \\
& \quad\quad\quad cut\_pt_{cut} \leftarrow \left(\underline{\mathbf{R}} + z \cdot m^{-1} \cdot \underline{\mathbf{e}}_{\underline{1}}\right) + r_w(z) \cdot m^{-1} \cdot \left(\underline{\mathbf{e}}_{\underline{2}} \cdot \cos\theta + \underline{\mathbf{e}}_{\underline{3}} \cdot \sin\theta\right) \\
& \quad\quad angles_{ang} \leftarrow cut\_pt \\
& \quad pins_{pin} \leftarrow angles \\
& pins
\end{aligned}
$$

Convert into pin error and sample at even intervals around the pin

```
rad_err := | for pin ∈ 1..number_of_pins
           |   for cut ∈ 1..number_of_cuts
           |     for ang ∈ 1..steps
           |       θ_ang ← angle([[[(cut_pt_1)_ang]_2]_1, [[(cut_pt_1)_ang]_2]_2])
           |       rad_err_ang ← sqrt([[[(cut_pt_pin)_ang]_cut]_1]^2 + [[[(cut_pt_pin)_ang]_cut]_2]^2) − r_c(θ_ang)·m^-1
           |     sorted ← csort(augment(θ, rad_err), 1)
           |     vs ← cspline(sorted<1>, sorted<2>)
           |     for ang ∈ 1..steps
           |       fit_ang ← interp(vs, sorted<1>, sorted<2>, a_ang)
           |     cuts_cut ← fit
           |   pins_pin ← cuts
           | pins
```

Calculate index of #1 pin

```
index_err_1 := | pin ← 1
               | cut ← 2
               | sum_of_y ← 0
               | for ang ∈ 1..steps
               |   sum_of_y ← sum_of_y + [[(rad_err_pin)_cut]_ang]·sin(a_ang)
               | index_err_1 ← 2·sum_of_y / steps
```

```
cut_pt := | cut_pt if index_compensation_on = FALSE
          | otherwise
          |   for pin ∈ 1..number_of_pins
          |     for ang ∈ 1..steps
          |       for cut ∈ 1..number_of_cuts
          |         rad ← [(rad_err_pin)_cut]_ang + r_c(a_ang)·m^-1
          |         new_pt ← [ rad·cos(a_ang)
          |                    rad·sin(a_ang) − index_err_1
          |                    [[(cut_pt_pin)_ang]_cut]_3 ]
          |         cuts_cut ← new_pt
          |       angles_ang ← cuts
          |     pins_pin ← angles
          |   pins
```

Convert into pin error and sample at even intervals around the pin

```
rad_err := | rad_err  if index_compensation_on = FALSE
           | otherwise
           |   | for pin ∈ 1..number_of_pins
           |   |   | for cut ∈ 1..number_of_cuts
           |   |   |   | for ang ∈ 1..steps
           |   |   |   |   | θ_ang ← angle([[(cut_pt)_ang]_2]_1, [[(cut_pt)_ang]_2]_2)
           |   |   |   |   | rad_err_ang ← sqrt([[[(cut_pt_pin)_ang]_cut]_1]^2 + [[[(cut_pt_pin)_ang]_cut]_2]^2) − r_c(θ_ang)·m^-1
           |   |   |   | sorted ← csort(augment(θ, rad_err), 1)
           |   |   |   | vs ← cspline(sorted<1>, sorted<2>)
           |   |   |   | for ang ∈ 1..steps
           |   |   |   |   fit_ang ← interp(vs, sorted<1>, sorted<2>, a_ang)
           |   |   |   | cuts_cut ← fit
           |   |   | pins_pin ← cuts
           |   | pins
```

Calculate index

```
index_err := | for pin ∈ 1..number_of_pins
             |   | for cut ∈ 1..number_of_cuts
             |   |   | sum_of_y ← 0
             |   |   | for ang ∈ 1..steps
             |   |   |   sum_of_y ← sum_of_y + [[(rad_err_pin)_cut]_ang]·sin(a_ang)
             |   |   | index_err ← 2·(sum_of_y / steps)
             |   |   | cuts_cut ← index_err
             |   | pins_pin ← cuts
             | pins
```

```
throw_err := | for pin ∈ 1..number_of_pins
             |   | for cut ∈ 1..number_of_cuts
             |   |   | sum_of_x ← 0
             |   |   | for ang ∈ 1..steps
             |   |   |   sum_of_x ← sum_of_x + [[(rad_err_pin)_cut]_ang]·cos(a_ang)
             |   |   | throw_err ← 2·sum_of_x / steps
             |   |   | cuts_cut ← throw_err
             |   | pins_pin ← cuts
             | pins
true_rad_err := | for pin ∈ 1..number_of_pins
                |   | for cut ∈ 1..number_of_cuts
                |   |   | for ang ∈ 1..steps
                |   |   |   | rad ← [(rad_err_pin)_cut]_ang + r_c(a_ang)·m^-1
                |   |   |   | v ← [rad·cos(a_ang) - (throw_err_pin)_cut ; rad·sin(a_ang) - (index_err_pin)_cut]
                |   |   |   | true_rad_err ← |v| - r_c(a_ang)·m^-1
                |   |   |   | angles_ang ← true_rad_err
                |   |   | cuts_cut ← angles
                |   | pins_pin ← cuts
                | pins
```

Place the error signature in a vector and write to a file

```
S := | for pin ∈ 1..1
     |   for cut ∈ 1..number_of_cuts
     |     for ang ∈ 1..steps
     |       output_((pin-1)·number_of_cutssteps+(cut-1)·steps+ang) ← [(rad_err_pin)_cut]_ang
     | output
WRITEPRN(output) := S
```

```
size_err := | for pin ∈ 1..number_of_pins
            |   | for cut ∈ 1..number_of_cuts
            |   |   | size_err ← 2·mean[(true_rad_err_pin)_cut]
            |   |   | cuts_cut ← size_err
            |   | pins_pin ← cuts
            | pins
taper := | for pin ∈ 1..number_of_pins
         |   | taper ← slope(cut_z, size_err_pin)
         |   | pins_pin ← taper
         | pins
roundness_err := | for pin ∈ 1..number_of_pins
                 |   | for cut ∈ 1..number_of_cuts
                 |   |   | r_min ← min[(true_rad_err_pin)_cut]
                 |   |   | r_max ← max[(true_rad_err_pin)_cut]
                 |   |   | roundness ← r_max − r_min
                 |   |   | cuts_cut ← roundness
                 |   | pins_pin ← cuts
                 | pins
```

F. Error Shapes for CNC Crankpin Grinding

This appendix is a set of graphs generated by the virtual machining model of the CNC crankpin grinding process in Appendix E. Each page in this appendix corresponds to a single noise factor. The noise factors modeled here are:

- Grinding wheel attrition
- Main bearing size error
- Main bearing size mismatch
- Cumulative lead error in the x axis
- Once-per-evolution lead error in the x axis
- Backlash (lost motion) in the x axis
- Deflection of the machine due to acceleration of the X axis carriage
- Misalignment (height) of the x axis carriage
- Z squareness of the x axis carriage
- X Parallelism of the work spindles with the z axis

For each noise factor, there are polar plots of radial error for each of the six crankpins ground on the machine. The legend for the plots is given in Figure F.1. The polar plots have been stripped of units to conceal the capabilities of the machine. The Roman numerals in the titles of each polar plot follow the numbering convention in Figure F.2. The top of each plot is the top dead center location for each crankpin (see Figure F.3).

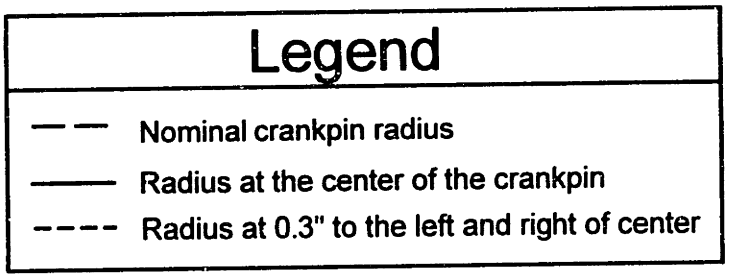


Figure F.1 Legend for the radial error plots.

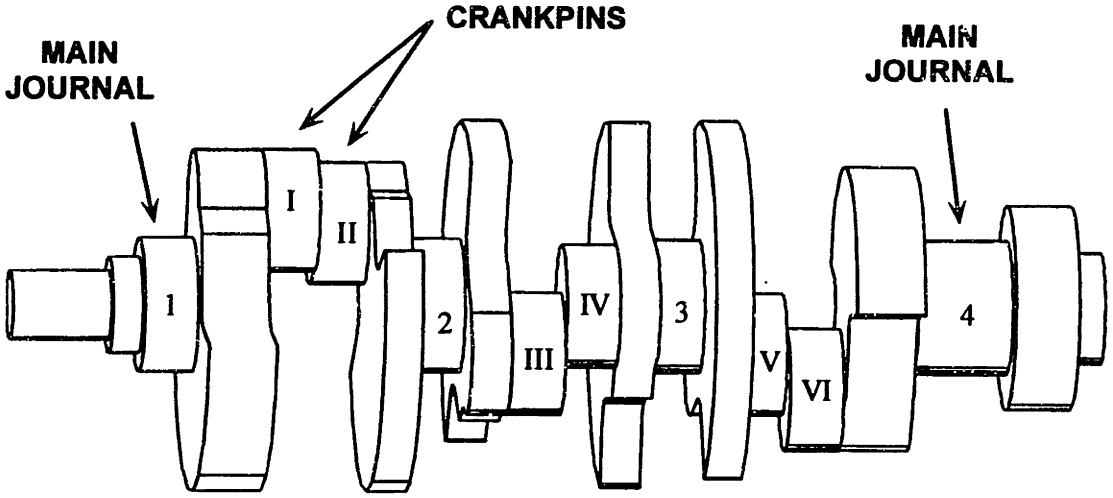


Figure F.2 A V6 engine crankshaft and associated numbering conventions.

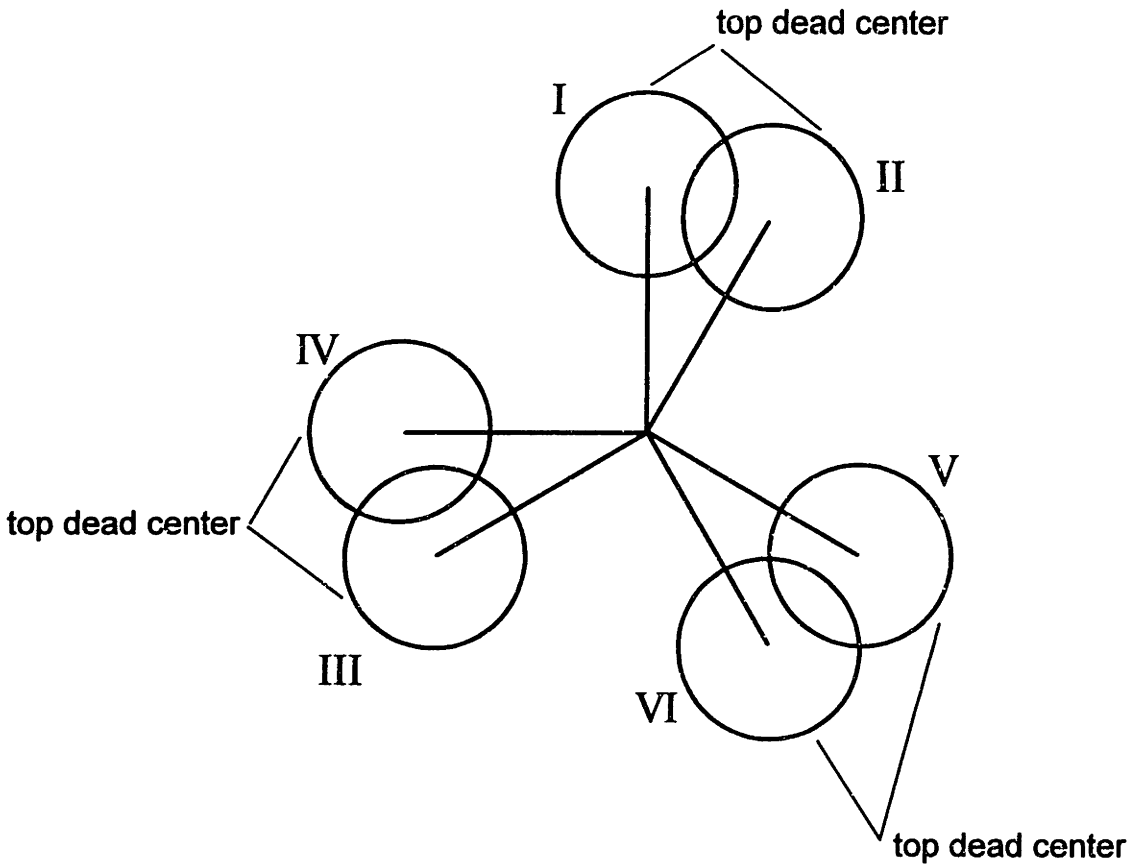
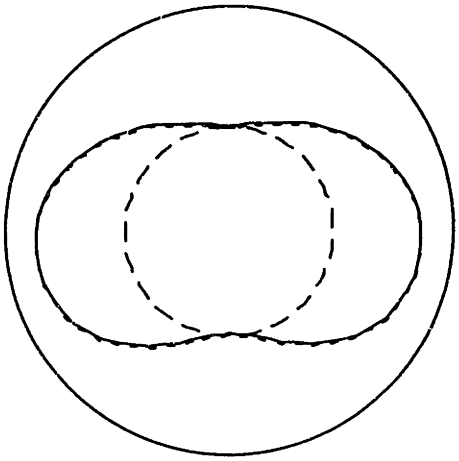
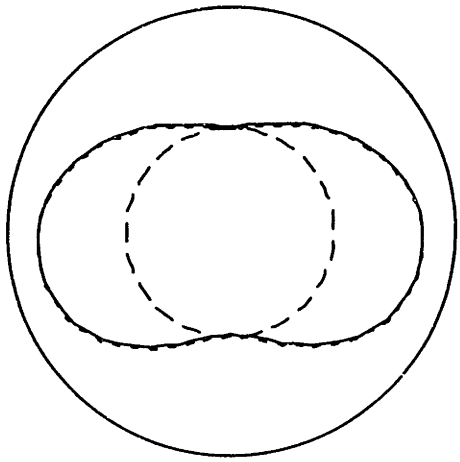


Figure F.3 The angular position of crankpins on a V6 engine crankshaft.

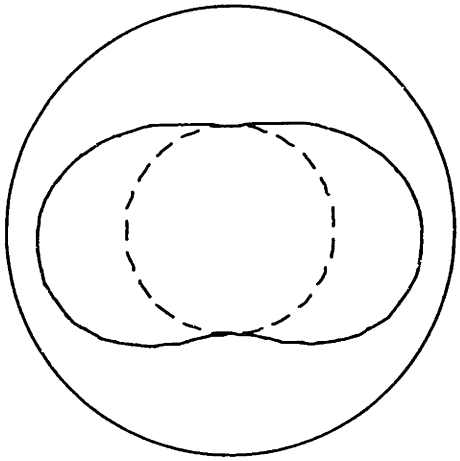
Grinding Wheel Attrition



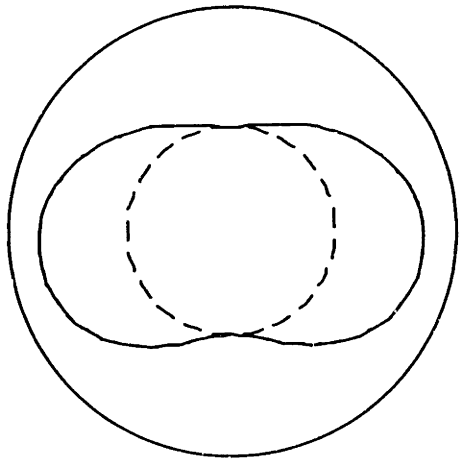
Pin I



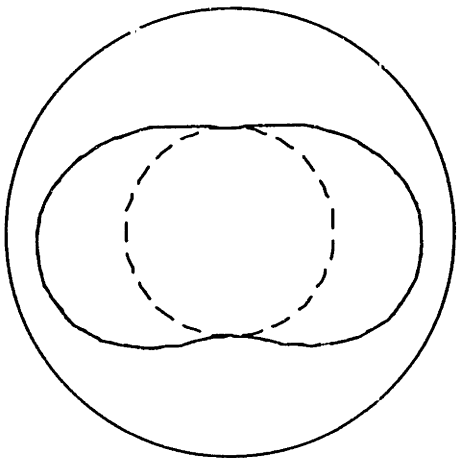
Pin II



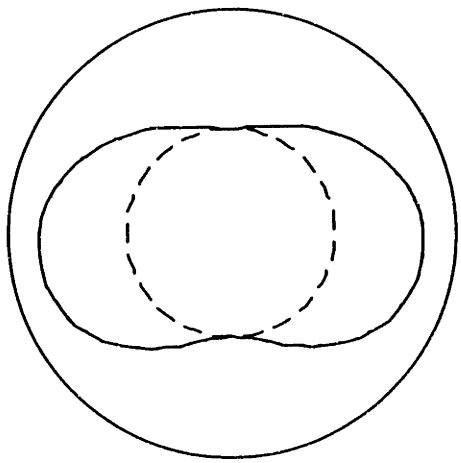
Pin III



Pin IV

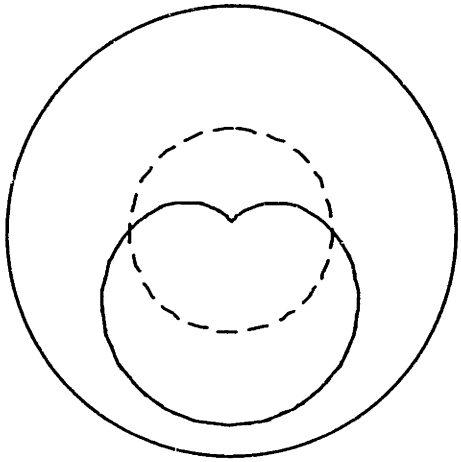


Pin V

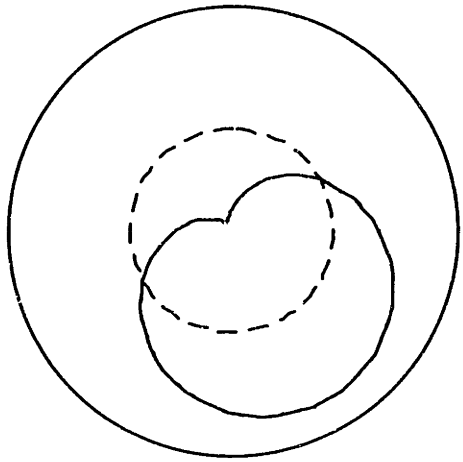


Pin VI

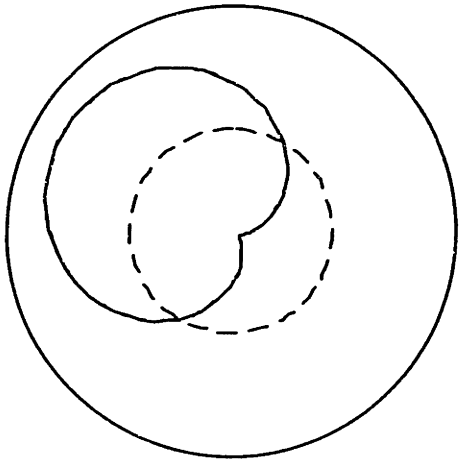
Main Bearing Size Error



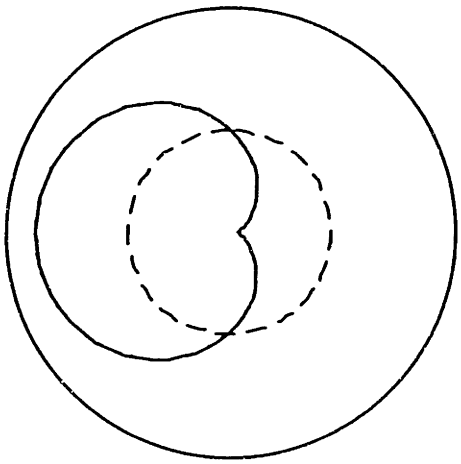
Pin I



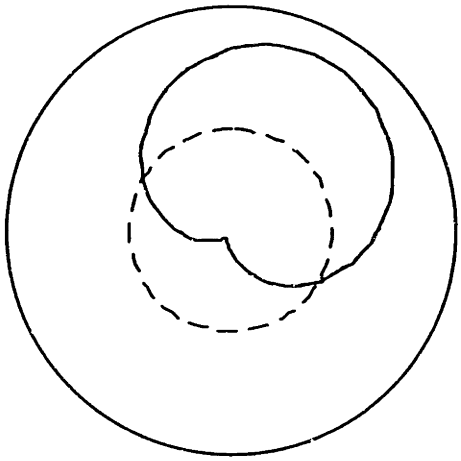
Pin II



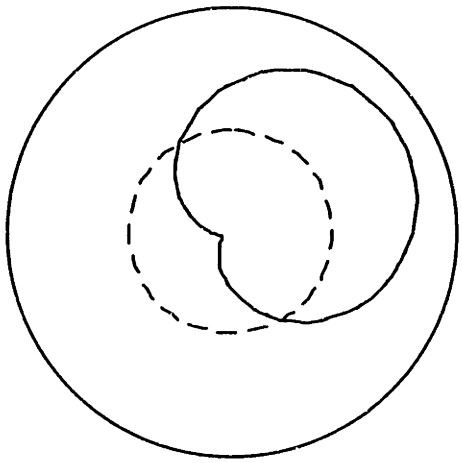
Pin III



Pin IV

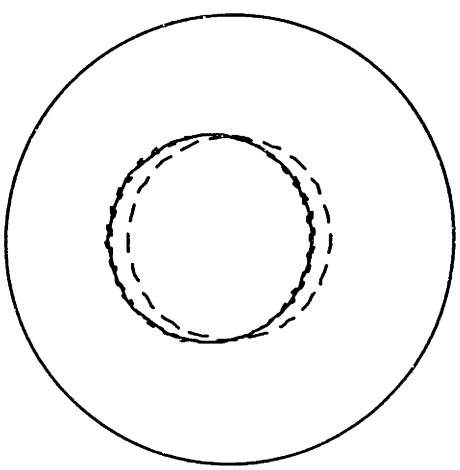


Pin V

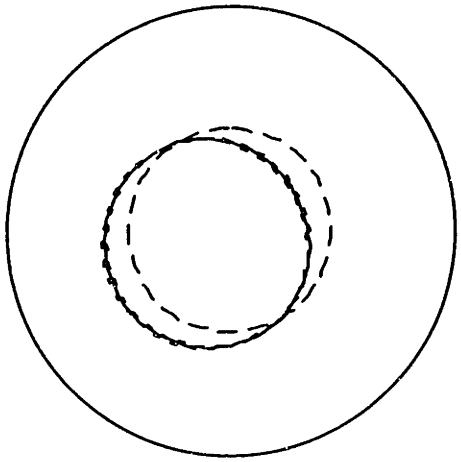


Pin VI

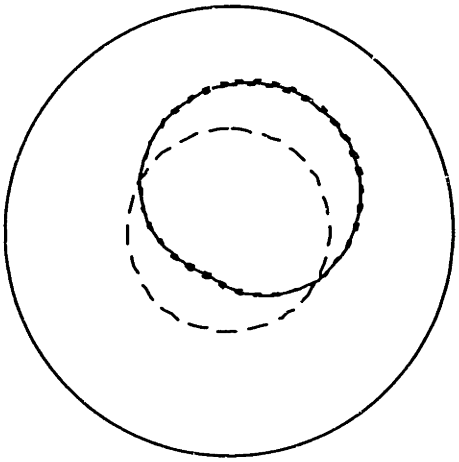
Main Bearing Size Mismatch



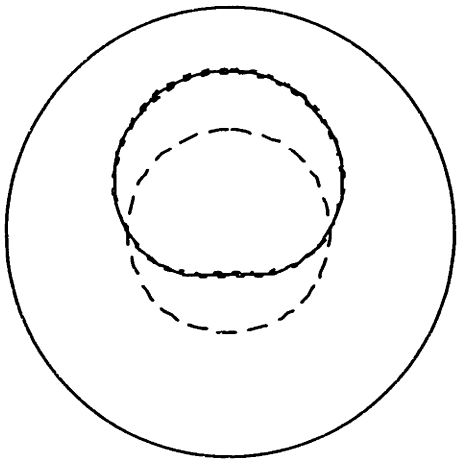
Pin I



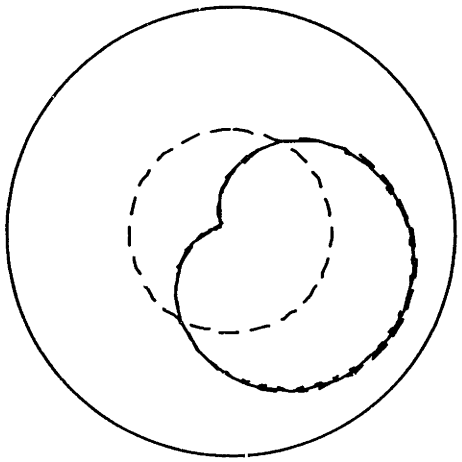
Pin II



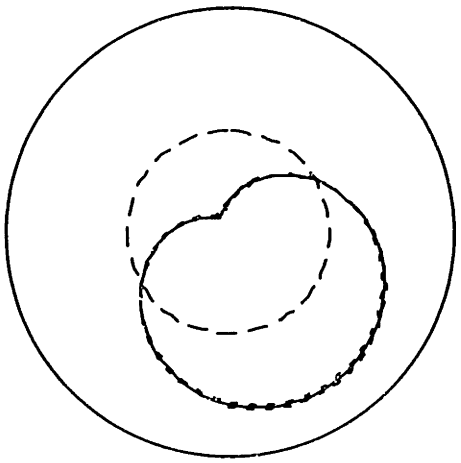
Pin III



Pin IV

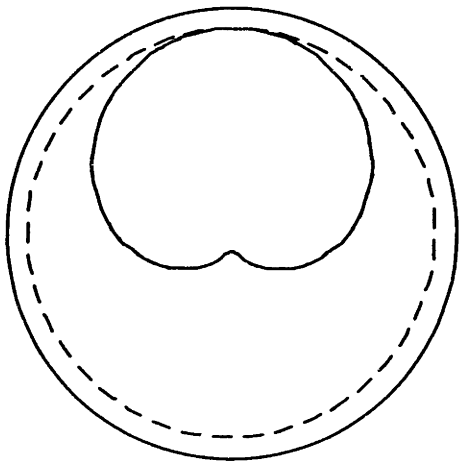


Pin V

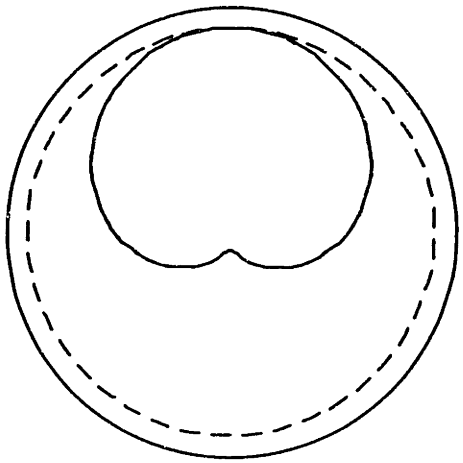


Pin VI

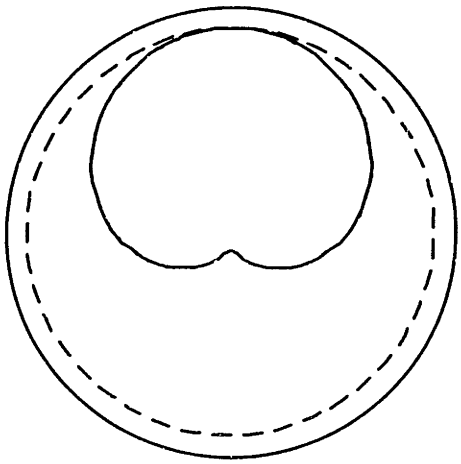
Cumulative Lead Error in the X Axis Lead Screw



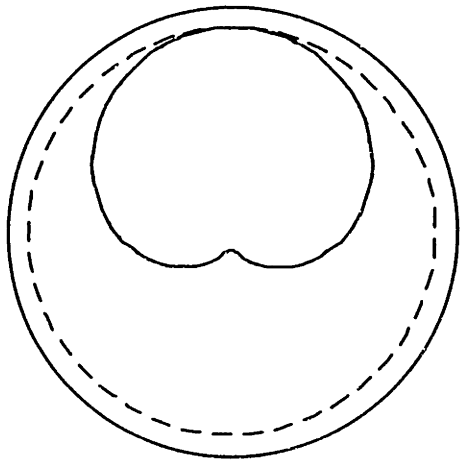
Pin I



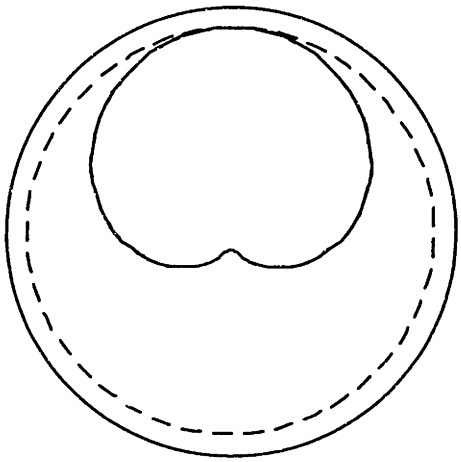
Pin II



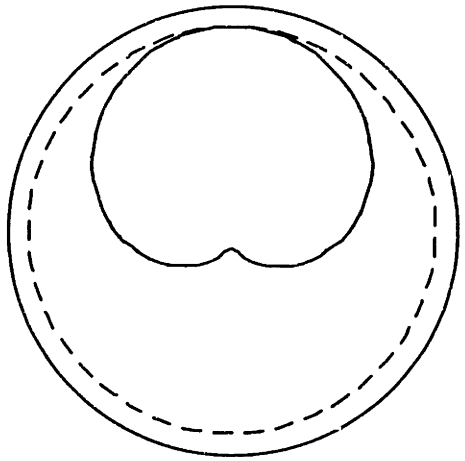
Pin III



Pin IV

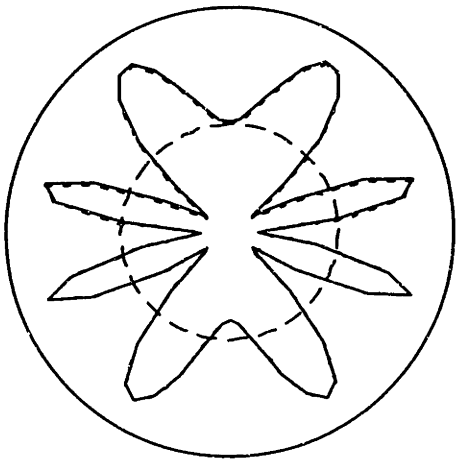


Pin V

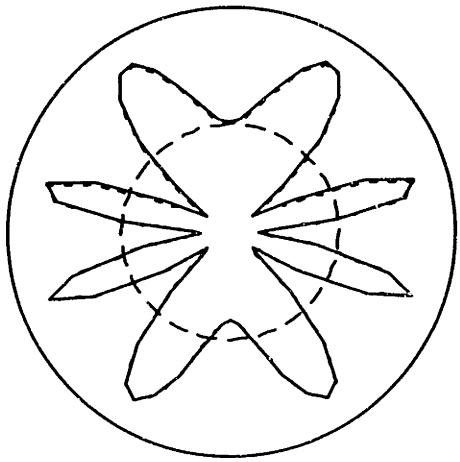


Pin VI

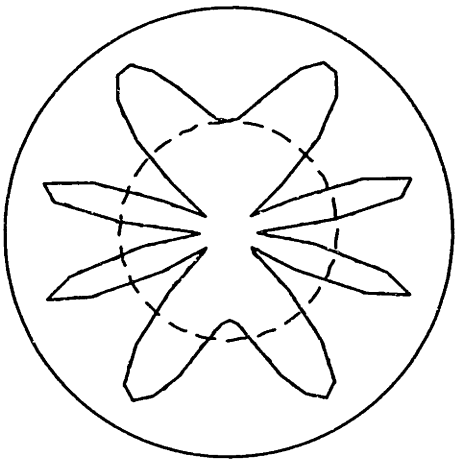
Once Per Revolution Error in the X Axis Lead Screw



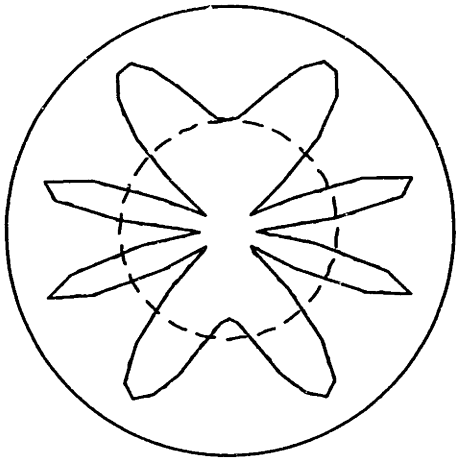
Pin I



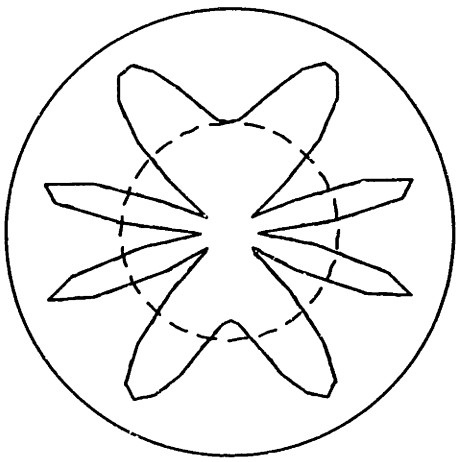
Pin II



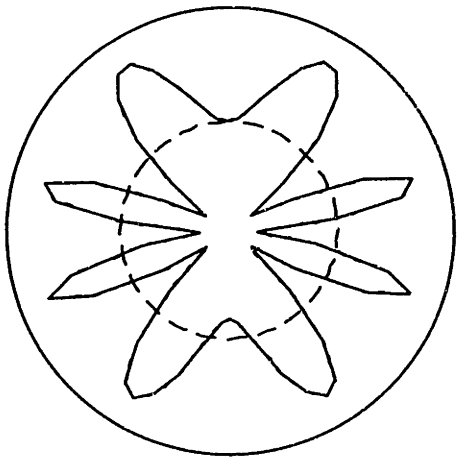
Pin III



Pin IV

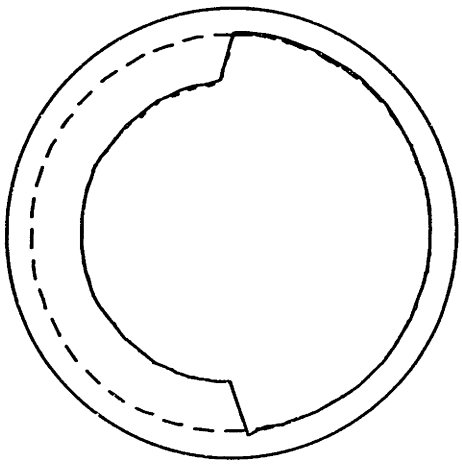


Pin V

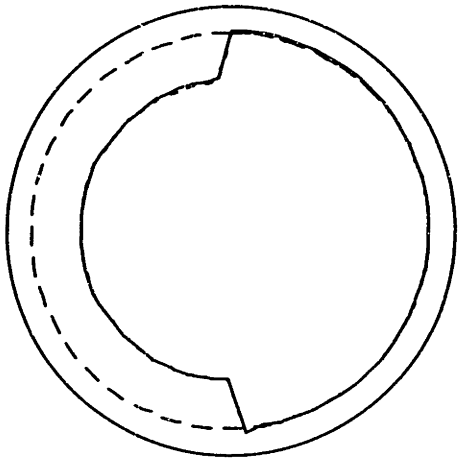


Pin VI

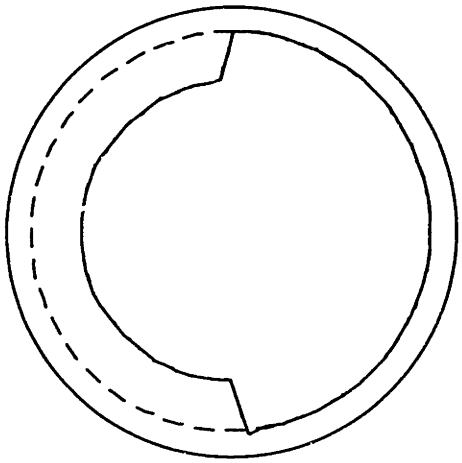
Backlash in the X Axis



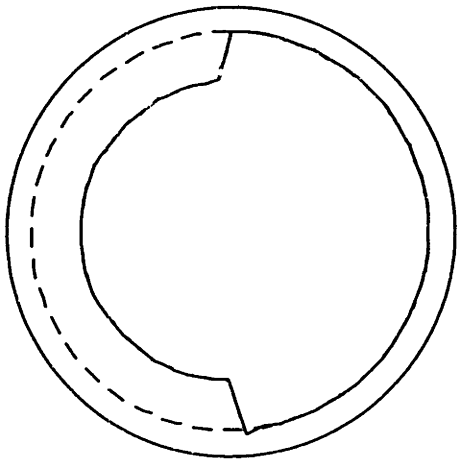
Pin I



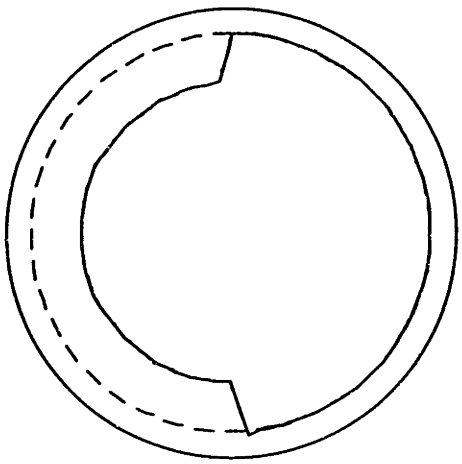
Pin II



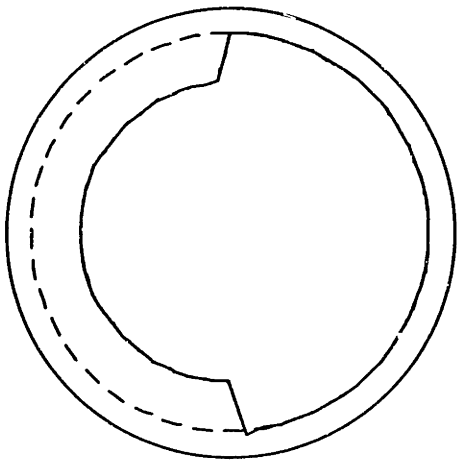
Pin III



Pin IV

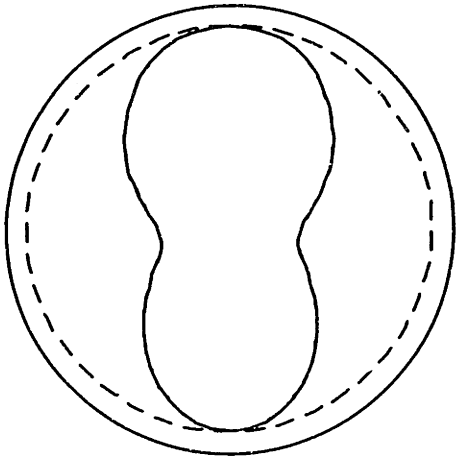


Pin V

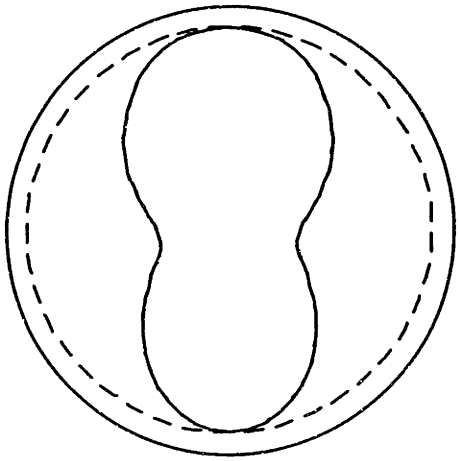


Pin VI

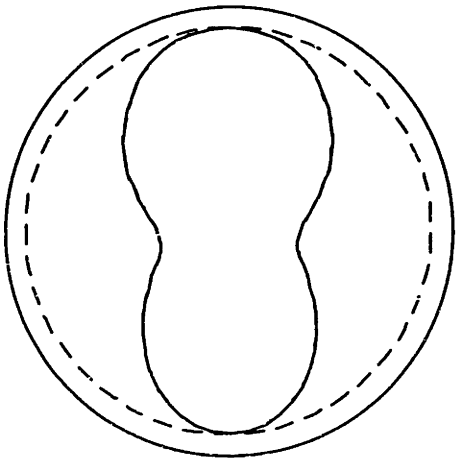
Deflection of the Machine Due to Acceleration of the X Axis Carriage



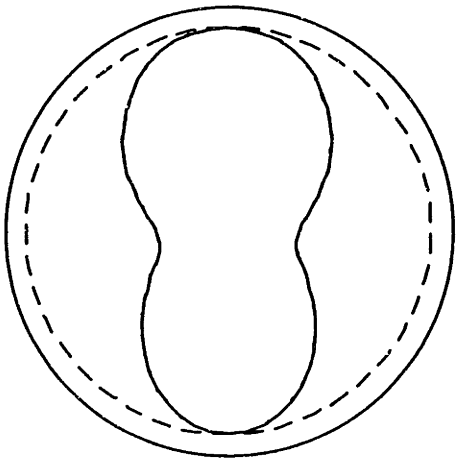
Pin I



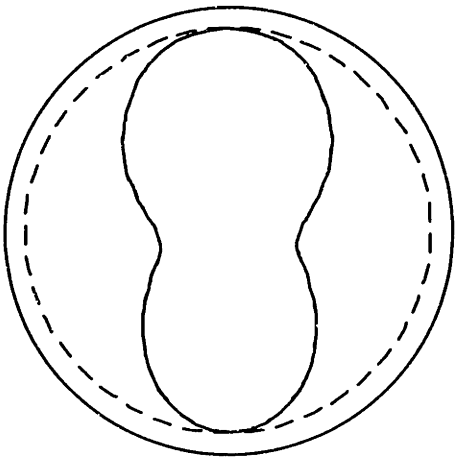
Pin II



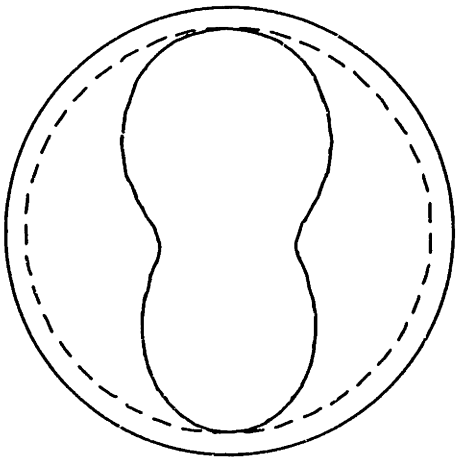
Pin III



Pin IV

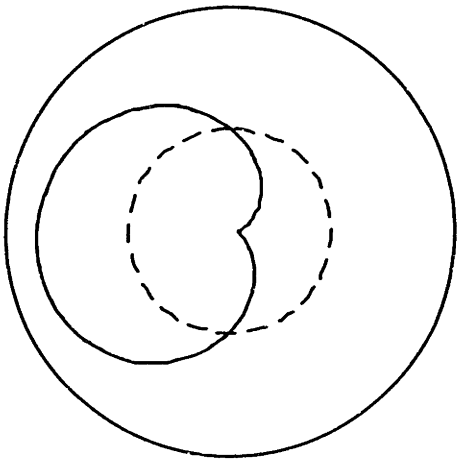


Pin V

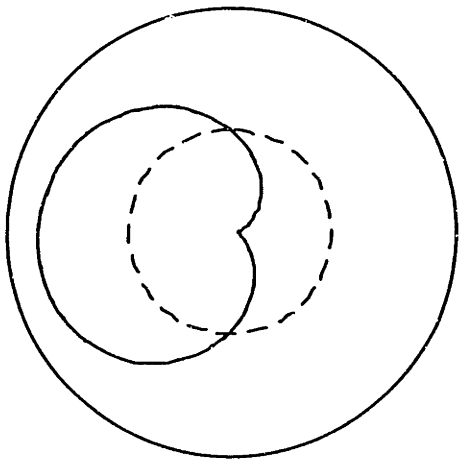


Pin VI

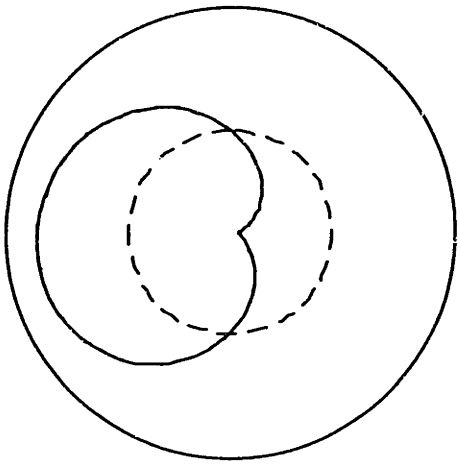
Misalignment (Height) of the X Axis Carriage



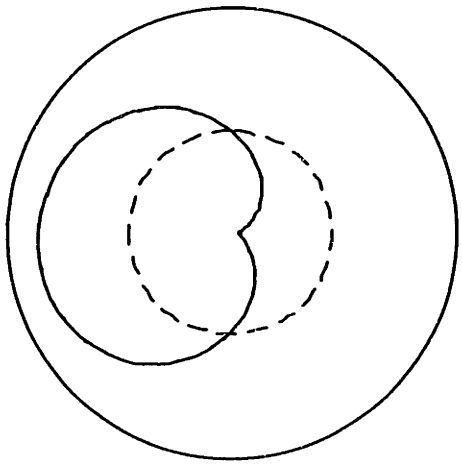
Pin I



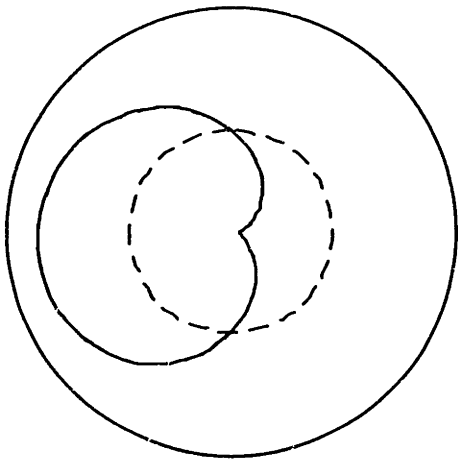
Pin II



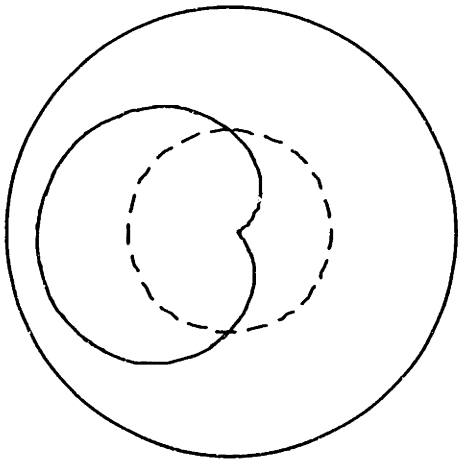
Pin III



Pin IV

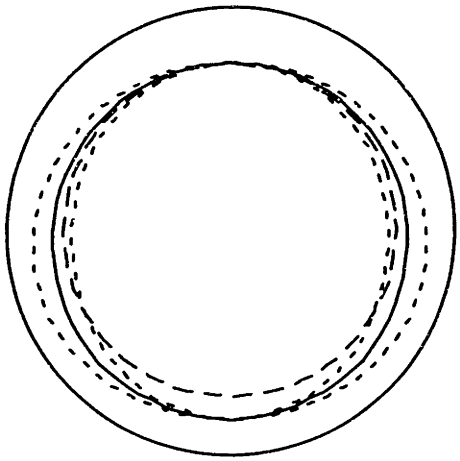


Pin V

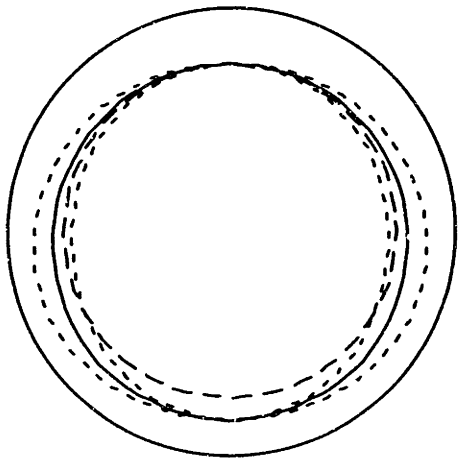


Pin VI

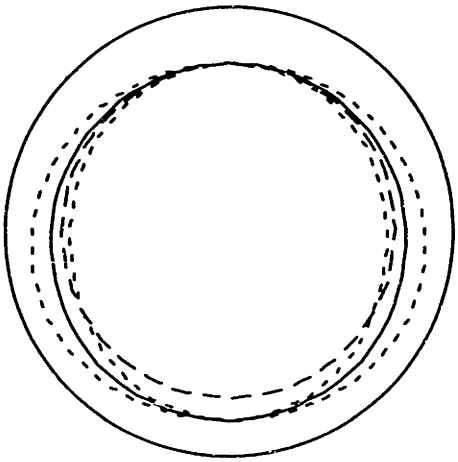
Z Squareness of the X Axis Carriage



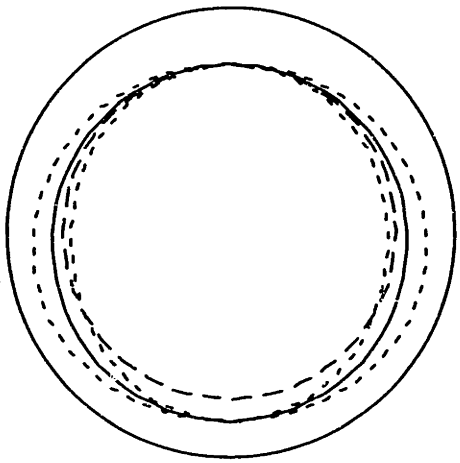
Pin I



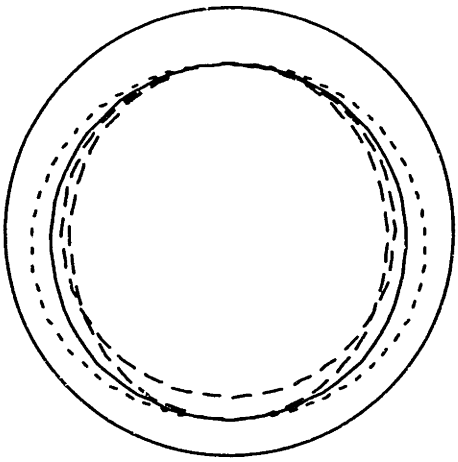
Pin II



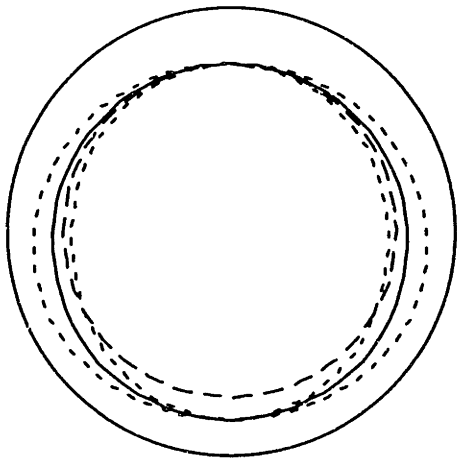
Pin III



Pin IV

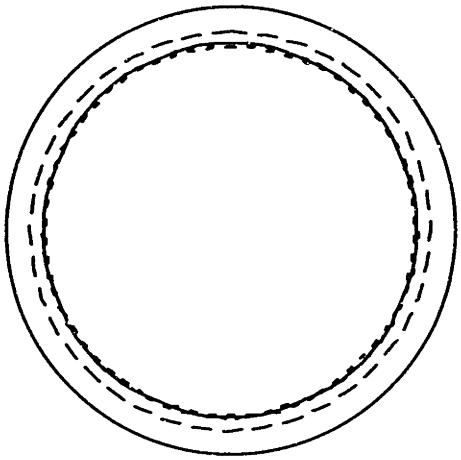


Pin V

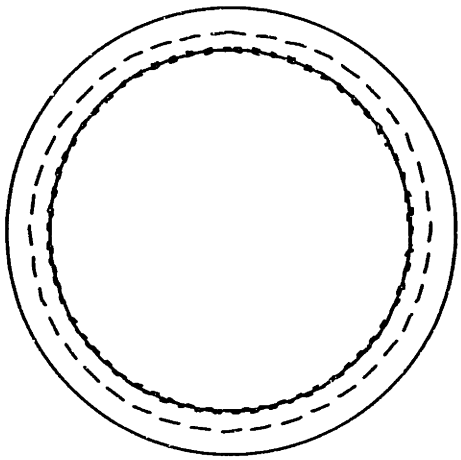


Pin VI

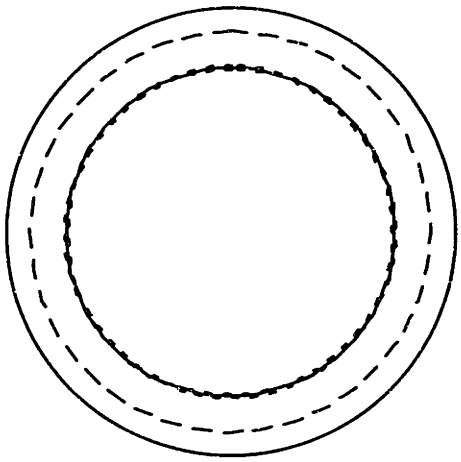
X Parallelism of the Work Spindles with the Z axis



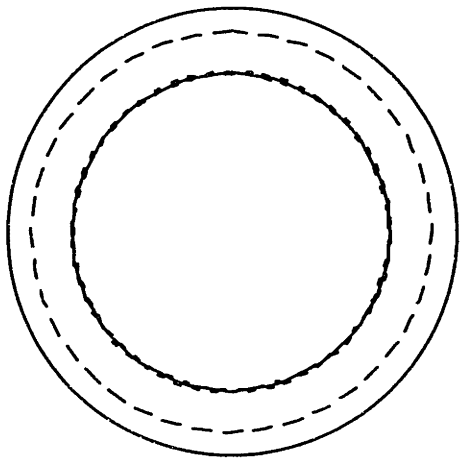
Pin I



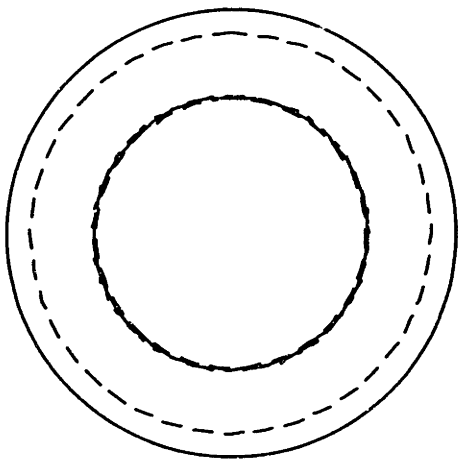
Pin II



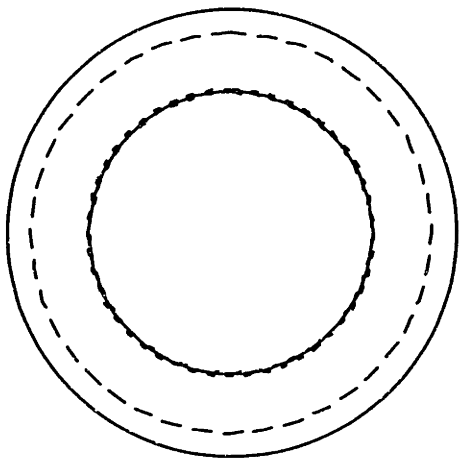
Pin III



Pin IV



Pin V

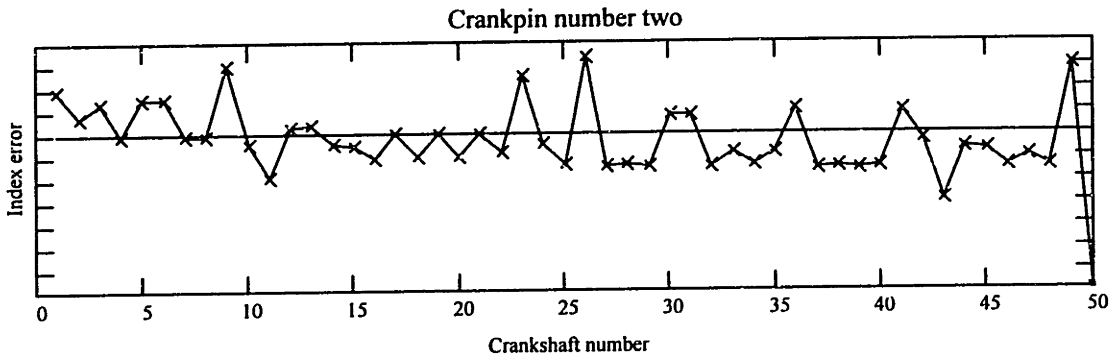


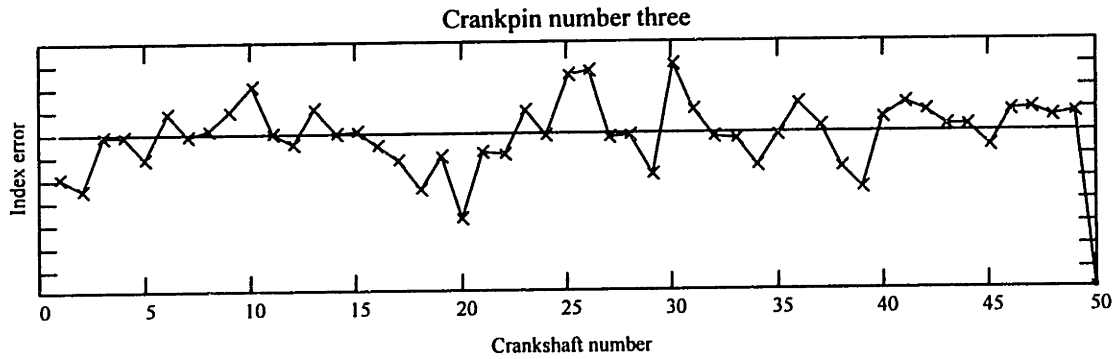
Pin VI

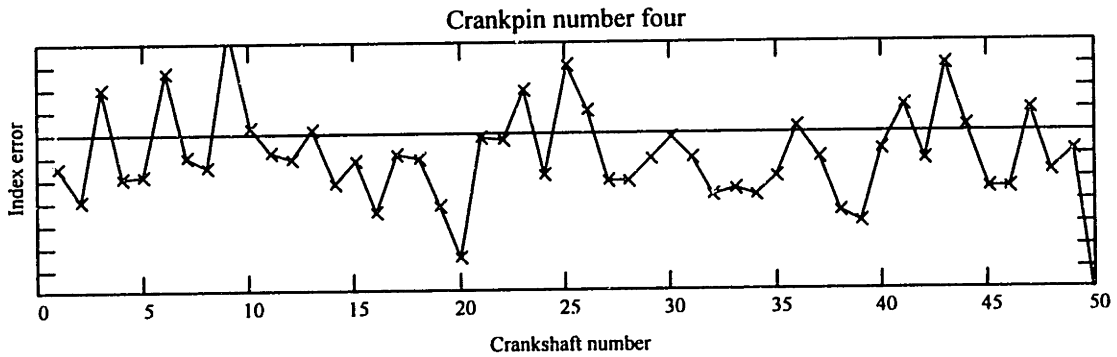
G. Data from CNC Crankpin Grinding

This appendix contains data from SPC charts for the CNC crankpin grinding process. A total of 98 crankshafts were ground, 50 on one machine and 48 on a second machine. The SPC data for the second machine are only presented here for a few quality characteristics where necessary to support the discussion in Chapter 11. The units and numerical values have been stripped form the ordinate of the graphs to mask the capability of the machines.

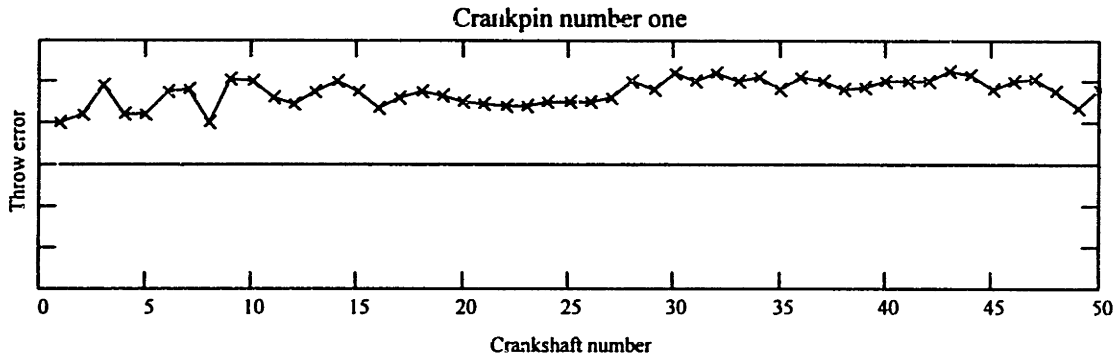
INDEX ERROR - MACHINE #1

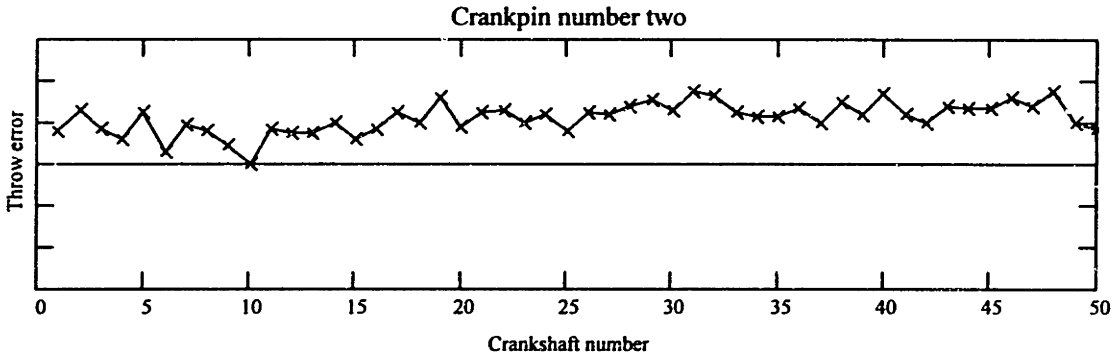


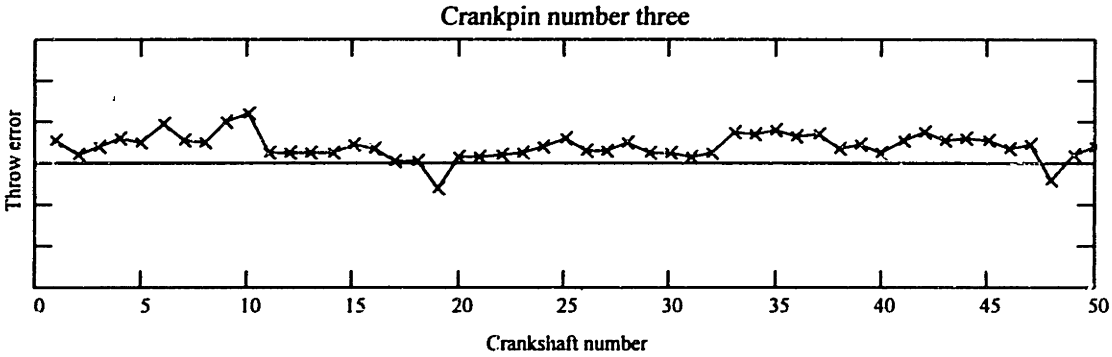


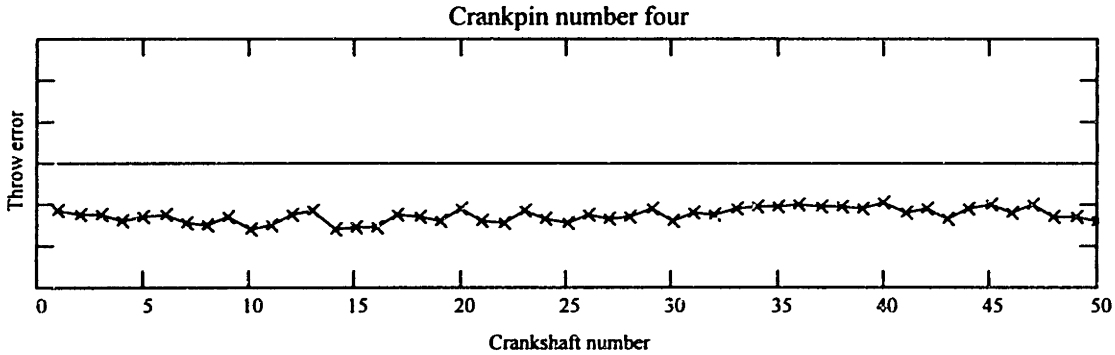


THROW ERROR - MACHINE #1

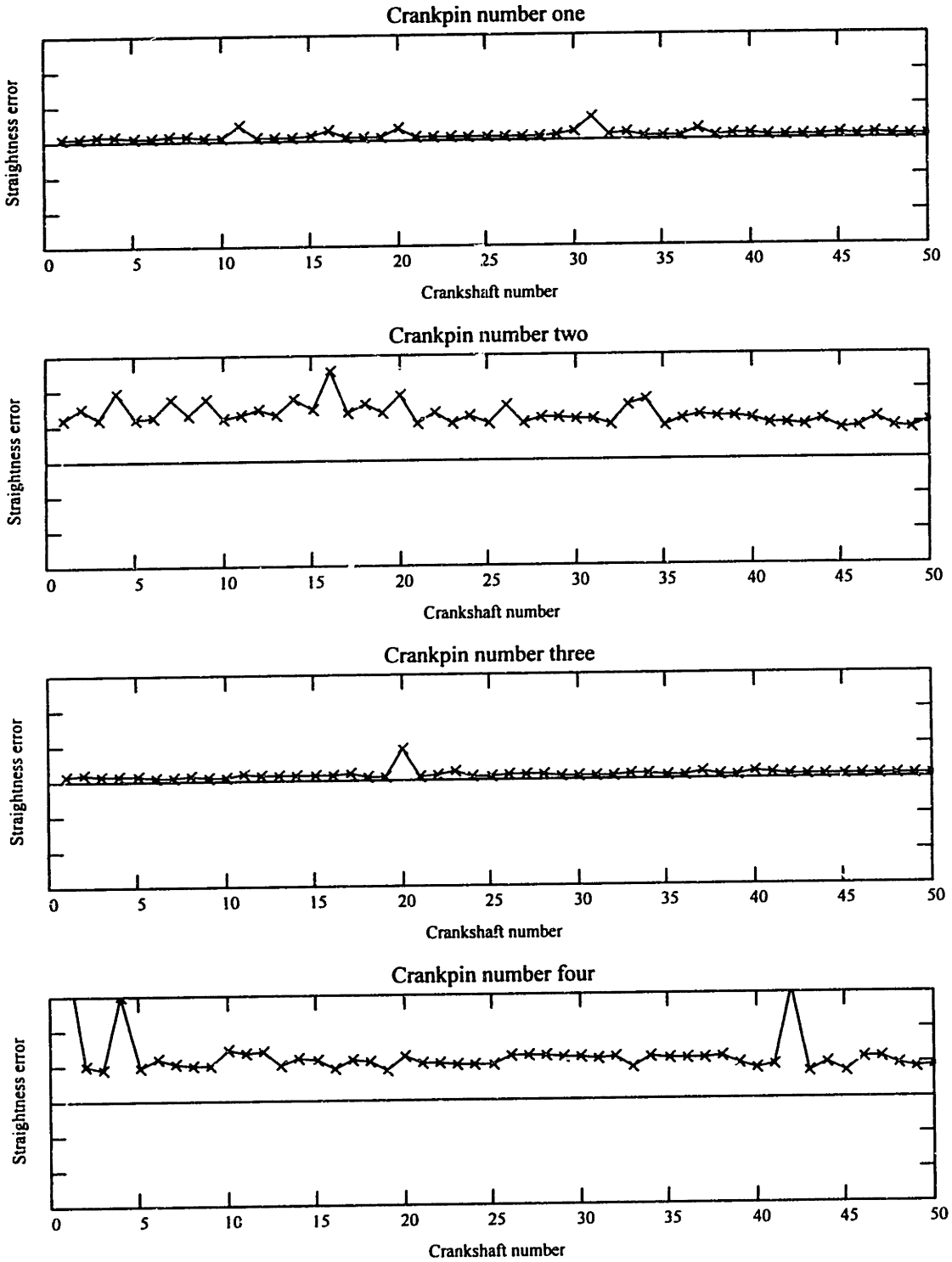




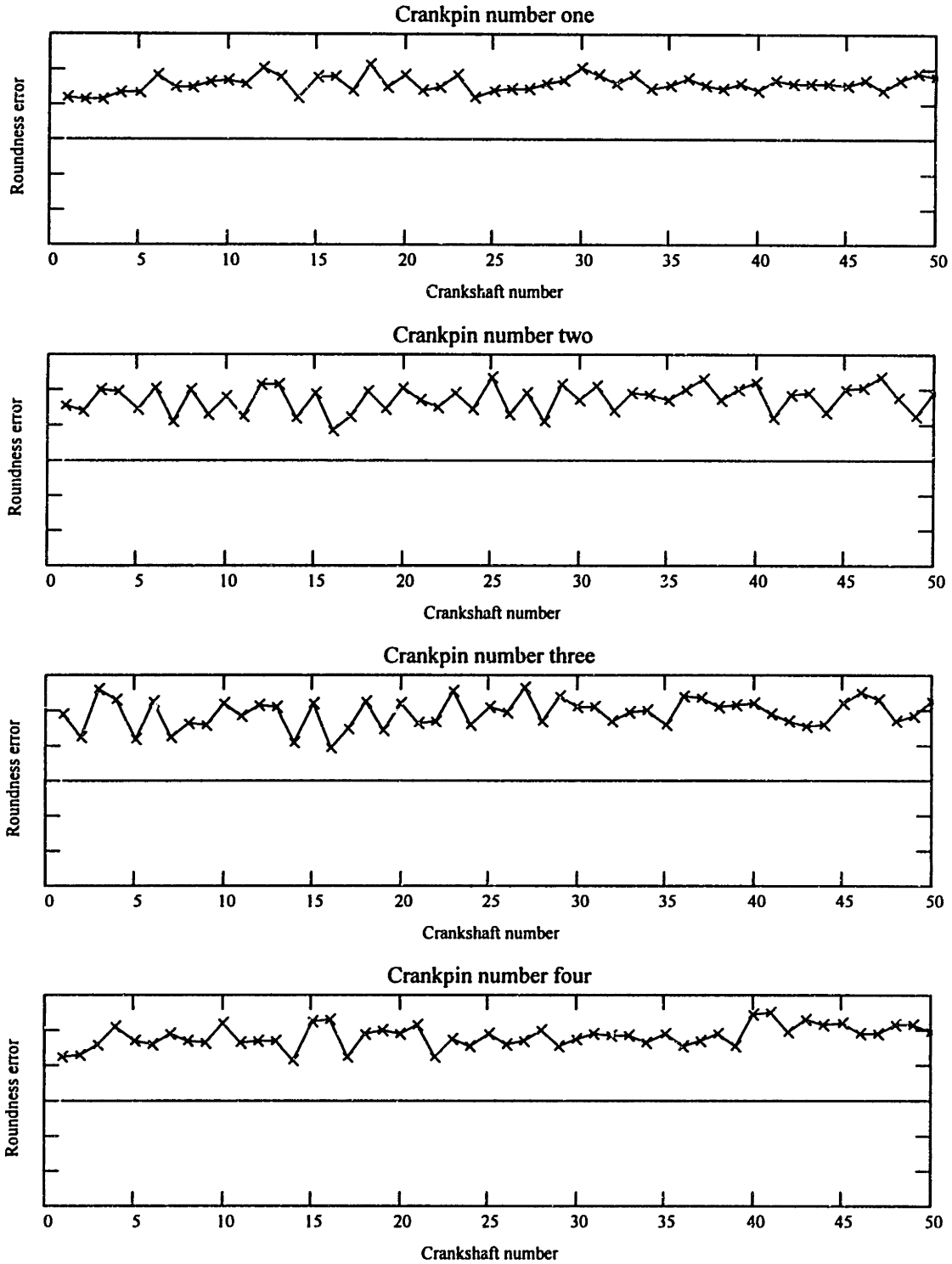




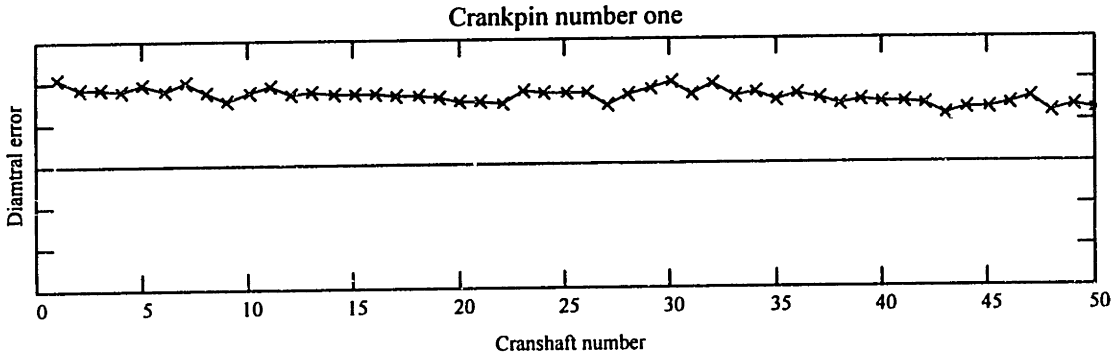
STRAIGHTNESS ERROR - MACHINE #1

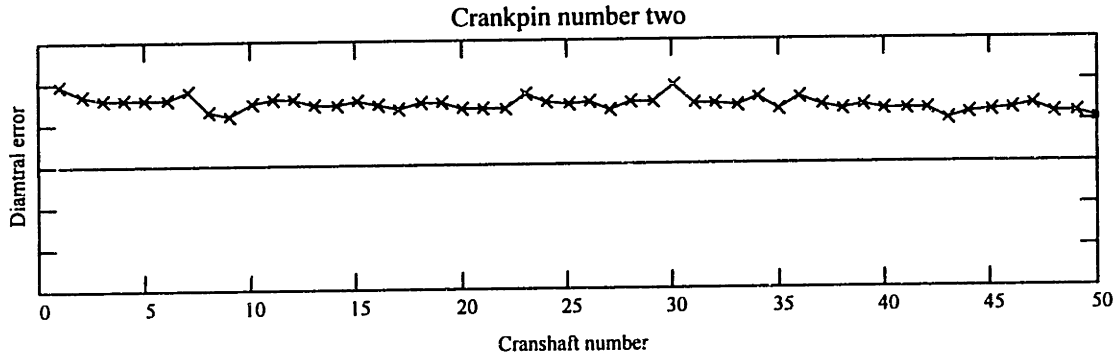


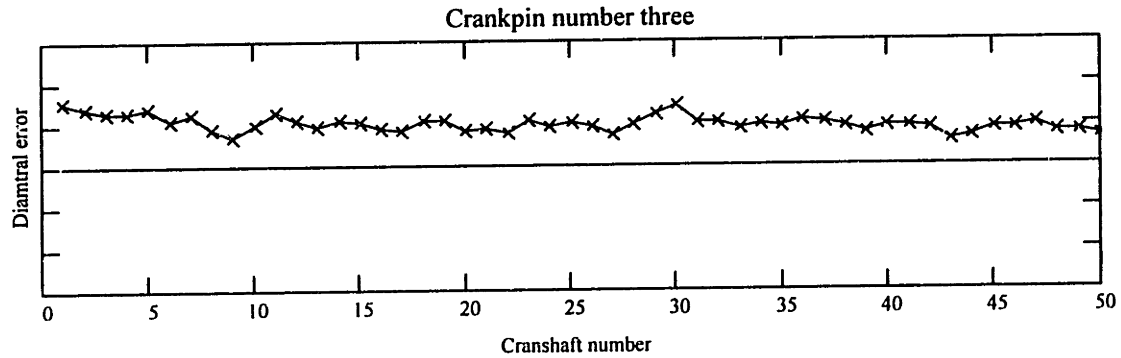
ROUNDNESS ERROR - MACHINE #1

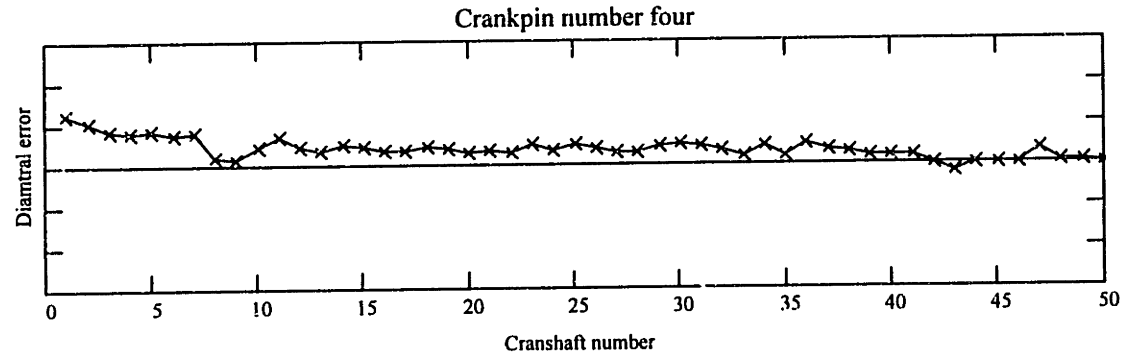


DIAMETRAL ERROR - MACHINE #1

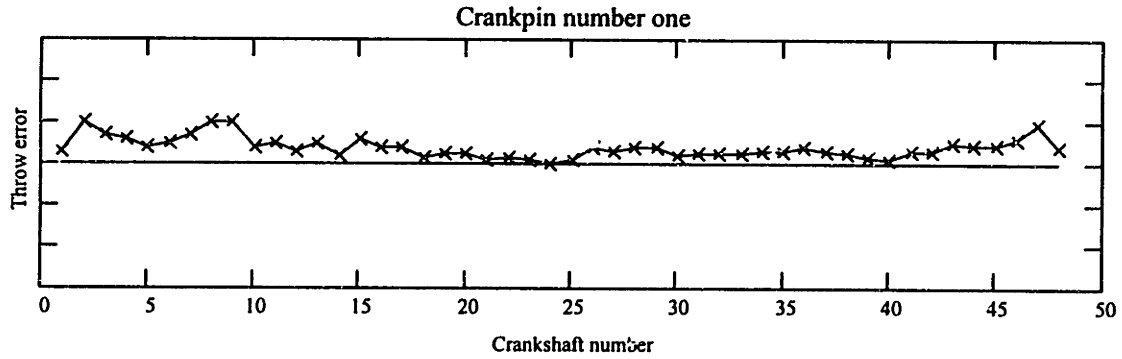




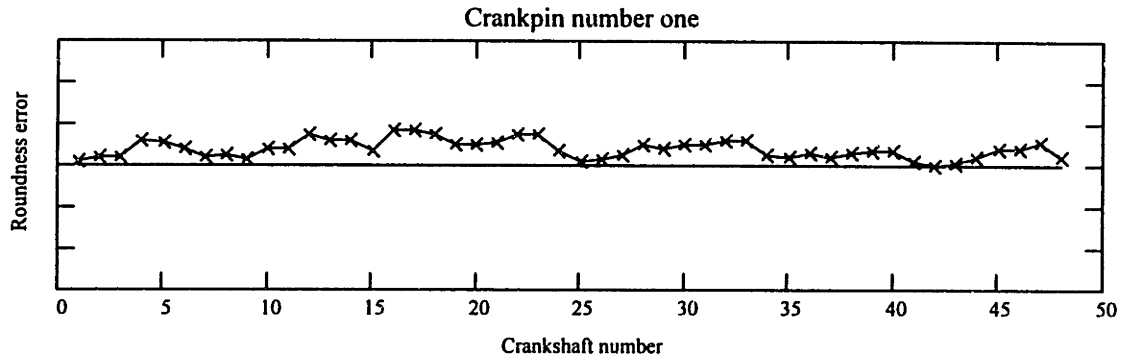




THROW ERROR - MACHINE #2



ROUNDNESS ERROR - MACHINE #2



H. Algorithm for Computing the Volume of a Polytope

This Appendix describes an algorithm I developed to compute the volume of a polytope. It is based on a theorem by Lasserre [1981]. Actually, Lasserre also described an algorithm for computing the volume, but it was designed for symbolic solution. This appendix documents my adaptation of the algorithm to numerical solution techniques and the tricks used to ensure generality applicability of the solution to and capability matrix and bias vector.

Given a convex polyhedron defined by the set of linear inequalities

$$\mathbf{Ax} \le \mathbf{b} \qquad \text{F.1}$$

the volume of polyhedron is

$$V(n,\mathbf{A},\mathbf{b}) = \frac{1}{n}\sum_{p=1}^{m}\frac{\mathbf{b}_p}{\left|\mathbf{A}_{p,q}\right|}\cdot V(n-1,\tilde{\mathbf{A}},\tilde{\mathbf{b}}) \qquad \text{F.2}$$

where $\tilde{\mathbf{A}}\mathbf{x} \le \tilde{\mathbf{b}}$ is the system resulting from removing $\mathbf{x}_q$ from the system $\mathbf{Ax} \le \mathbf{b}$ by casting the pth inequality as an equality [Lasserre, 1981].

The reduced system $\tilde{\mathbf{A}}\mathbf{x} \le \tilde{\mathbf{b}}$ can be expressed as

$$\mathbf{A}_p\mathbf{x} = \mathbf{b}_p \text{ and } \tilde{\mathbf{A}}\begin{Bmatrix}\mathbf{x}_1\\ \vdots \\ \mathbf{x}_{q-1}\\ \mathbf{x}_{q+1}\\ \vdots \\ \mathbf{x}_n\end{Bmatrix} \le \tilde{\mathbf{b}} \qquad \text{F.3}$$

where

$$\tilde{\mathbf{A}}_{i,j} = \mathbf{A}_{\begin{bmatrix} i \text{ if } i<p \\ i+1 \text{ otherwise}\end{bmatrix},\begin{bmatrix} j \text{ if } j<q \\ j+1 \text{ otherwise}\end{bmatrix}} - \mathbf{A}_{\begin{bmatrix} i \text{ if } i<p \\ i+1 \text{ otherwise}\end{bmatrix},q}\frac{\mathbf{A}_{p,\begin{bmatrix} j \text{ if } j<q \\ j+1 \text{ otherwise}\end{bmatrix}}}{\mathbf{A}_{p,q}} \qquad \text{F.4}$$

and

$$\tilde{\mathbf{b}}_{\begin{bmatrix} i \text{ if } i<p \\ i+1 \text{ otherwise}\end{bmatrix}} = \mathbf{b}_{\begin{bmatrix} i \text{ if } i<p \\ i+1 \text{ otherwise}\end{bmatrix}} - \mathbf{A}_{\begin{bmatrix} i \text{ if } i<p \\ i+1 \text{ otherwise}\end{bmatrix},q}\frac{\mathbf{b}_p}{\mathbf{A}_{p,q}} \qquad \text{F.5}$$

where $i \in 1\ldots 2(m-1)$, $\mathrm{j} \in 1\ldots 2(\mathrm{n}-1)$.

The implementation of the recursive algorithm defined by Equation F.2 requires some care in implementation. On each evaluation of $V(n,\mathbf{A},\mathbf{b})$, any duplicate inequality constraints must be removed. That is, if two rows of **A** are identical and the corresponding elements of **b** are also the same, then one of the rows must be removed from the system. If not, the algorithm will sum the volume of a single face more than once. It also requires care in selecting which $\mathbf{x}_q$ to remove for any given i since if $\mathbf{A}_{i,q}$ is zero (or very small) an overflow will result.

A coding of the recursive algorithm follows. The first two arguments of the function V(m,n,A) is the number of rows and columns respectively in the matrix **A**. The third argument is the matrix **A** and vector **b** augmented into a single matrix $[\mathbf{A}\ \mathbf{b}]$

Note that if the polyhedron is unbounded, then the function will return a very large number. This cannot occur in computing the rolled throughput yield as defined in Section 5.10. Also note that if the volume of the ith face as computed by the algorithm below is zero, then the ith constraint is redundant.

$$
\begin{array}{l}
V(m,n,A) := \left|\begin{array}{l}
\text{if } n>1 \\
\quad \left|\begin{array}{l}
\text{for } i \in 2..m \\
\quad \left|\begin{array}{l}
\text{same} \leftarrow \text{not}\left[\prod_{i2=1}^{i-1} \text{not}\left[\left[\left[\left(A^T\right)^{\langle i \rangle}\right]\cdot\left|\left(A^T\right)^{\langle i2 \rangle}\right| = \left[\left(A^T\right)^{\langle i2 \rangle}\right]\cdot\left|\left(A^T\right)^{\langle i \rangle}\right|\right]\cdot\left|\left(A^T\right)^{\langle i2 \rangle}\right|\cdot\left|\left(A^T\right)^{\langle i \rangle}\right|\right]\right] \\
\left(\begin{array}{l}\text{for } j \in 1..n+1 \\ A_{i,j} \leftarrow 0\end{array}\right) \text{ if same} \\
\left[\frac{1}{n}\cdot\left[\sum_{i=1}^{m}\left[\left|\begin{array}{l}
\text{for } j\_elim \in 1..n \\
\quad \text{break if } A_{i,j\_elim} \neq 0 \\
0 \text{ if } A_{i,j\_elim} = 0 \\
\text{otherwise} \\
\quad \left|\begin{array}{l}
\text{for } i2 \in 1..m \\
\quad \left|\begin{array}{l}
Ap_{i2,n} \leftarrow A_{i2,n+1} - \left(A_{i2,j\_elim}\cdot\frac{A_{i,n+1}}{A_{i,j\_elim}}\right) \\
\text{for } j \in 1..n-1 \\
Ap_{i2,j} \leftarrow \left[A_{i2,\left[\left|\begin{array}{l} j \text{ if } j<j\_elim \\ (j+1) \text{ otherwise}\end{array}\right|\right]} - A_{i2,j\_elim}\cdot\frac{A_{i,\left[\left|\begin{array}{l} j \text{ if } j<j\_elim \\ (j+1) \text{ otherwise}\end{array}\right|\right]}}{A_{i,j\_elim}}\right]
\end{array}\right. \\
\left(\frac{A_{i,n+1}}{\left|A_{i,j\_elim}\right|}\cdot V(m,n-1,Ap)\right) \text{ if } A_{i,j\_elim} \neq 0 \\
0 \text{ otherwise}
\end{array}\right.
\end{array}\right|\right]\right]\right]
\end{array}\right. \\
\text{otherwise} \\
\quad \left|\begin{array}{l}
EMPTY \leftarrow 0 \\
\text{for } L \in 1..m \\
\quad \left|\begin{array}{l}
(EMPTY \leftarrow 1) \text{ if } \left(A_{L,1}=0\right)\cdot\left(A_{L,2}<0\right) \\
pos_L \leftarrow \left[\begin{array}{l} \frac{A_{L,2}}{A_{L,1}} \text{ if } A_{L,1}>0 \\ 10^{10} \text{ otherwise}\end{array}\right] \\
neg_L \leftarrow \left[\begin{array}{l} \frac{A_{L,2}}{A_{L,1}} \text{ if } A_{L,1}<0 \\ -10^{10} \text{ otherwise}\end{array}\right]
\end{array}\right. \\
\max\left(\left(\begin{array}{c} 0 \\ \min(pos) - \max(neg) \end{array}\right)\right)\cdot \text{not}(EMPTY)
\end{array}\right.
\end{array}\right.
\end{array}
$$