Erratum
to
ASME B89.4.10-2000
Methods for Performance Evaluation of Coordinate Measuring System Software

On page 8, para. 5.5.1, last paragraph, second sentence is revised. The corrected sentence is presented below.

> Specifically, in each size category, one ratio shall be 0.02, two shall be 0.1, two shall be 0.3, two shall be 1, two shall be 3, and one shall be 10.

THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
Three Park Avenue, New York, NY 10016-5990

September 2003

L7500E

ASME B89.4.10-2000

METHODS FOR PERFORMANCE EVALUATION OF COORDINATE MEASURING SYSTEM SOFTWARE

AN AMERICAN NATIONAL STANDARD

The American Society of
Mechanical Engineers

The American Society of Mechanical Engineers

A N A M E R I C A N N A T I O N A L S T A N D A R D

METHODS FOR PERFORMANCE EVALUATION OF COORDINATE MEASURING SYSTEM SOFTWARE

ASME B89.4.10-2000

Date of Issuance: July 15, 2002

The next edition of this Standard is scheduled for publication in 2005. There will be no addenda issued to this Edition.

ASME issues written replies to inquiries concerning interpretations of technical aspects of this Standard.

The American Society of Mechanical Engineers
Three Park Avenue, New York, NY 10016-5990

CONTENTS

Tables

Mandatory Appendix

Nonmandatory Appendices

FOREWORD

Coordinate measuring systems (CMSs) rely upon software that processes coordinate data; often this software computes fits of geometric elements to such data. The performance of these fits can vary among software packages, and in some cases can be a significant contributor to the overall uncertainty of measurement.

The purpose of this document is to provide guidelines for evaluating the quality of solutions generated by CMS software and to define minimal documentation requirements for software providers. This Standard is concerned with testing the behavior of algorithm implementation, not the testing of algorithms themselves. It is not the intent of this document to endorse or rate any computational method or system. A mechanism for generating collections of test data sets is specified. While a specific, static collection of standardized test data sets is not defined, the generating mechanism can produce several collections of similar character.

This Standard was approved by the American National Standards Institute on December 1, 2000.

ASME STANDARDS COMMITTEE B89
Dimensional Metrology

(The following is the roster of the Committee at the time of approval of this Standard.)

OFFICERS

R. B. Hook, *Chair*
B. Parry, *Vice Chair*
P. Esteban, *Secretary*

COMMITTEE PERSONNEL

K. L. Blaedel, University of California
J. B. Bryan, Bryan Associates
T. Carpenter, U.S. Air Force
T. Charlton, Brown & Sharpe Manufacturing
P. Esteban, The American Society of Mechanical Engineers
G. A. Hetland, Hutchinson Technology
R. J. Hocken, University of North Carolina
R. B. Hook, Metcon
M. Liebers, Professional Instruments
B. Parry, Boeing Co.
B. R. Taylor, Renishaw PLC
R. C. Veale, National Institute of Standards and Technology

SUBCOMMITTEE 4: COORDINATE MEASURING TECHNOLOGY

R. B. Hook, *Chair,* Metcon
R. D. Donaldson, *Vice Chair,* Robert Donaldson Metrology
W. L. Beckwith, Jr., Brown & Sharpe Manufacturing
T. Carpenter, U.S. Air Force
T. Charlton, Jr., Charlton Associated
R. J. Hocken, University of North Carolina
J. A. Jalkio, University of St. Thomas
B. Parry, Boeing Co.
S. D. Phillips, National Institute of Standards and Technology
B. R. Taylor, Renishaw PLC
R. C. Veale, National Institute of Standards and Technology

WORKING GROUP 4.10: SOFTWARE EVALUATION

A. Griggs, *Chair,* Brown & Sharpe Manufacturing
M. T. Gale, Giddings & Lewis Measuring Systems
W. Gehner, Deere & Co.
R. J. Hocken, University of North Carolina
T. H. Hopp, National Institute of Standards and Technology
C. Leland, Deere & Co.
J. Lifson, GE Aircraft Engines

L. R. Maggiano, MTI Engineering Corp./Mitutoyo
W. McLendon, Lockheed Aeronautical Systems Company
K. J. Moritz, Raytheon
W. H. Rasnick, Martin Marietta Energy Systems
R. M. Roterdam, MTI Engineering Corp./Mitutoyo
C. M. Shakarji, National Institute of Standards and Technology
P. D. Thomas, MTI Corp.

CORRESPONDENCE WITH THE B89 COMMITTEE

General. ASME Codes and Standards are developed and maintained with the intent to represent the consensus of concerned interests. As such, users of this Standard may interact with the Committee by requesting interpretations, proposing revisions, and attending Committee meetings. Correspondence should be addressed to:

Secretary, B89 Main Committee
The American Society of Mechanical Engineers
Three Park Avenue
New York, NY 10016

Proposed Revisions. Revisions are made periodically to the standard to incorporate changes that appear necessary or desirable, as demonstrated by the experience gained from the application of the standard. Approved revisions will be published periodically.

The Committee welcomes proposals for revisions to this Standard. Such proposals should be as specific as possible: citing the paragraph number(s), the proposed wording, and a detailed description of the reasons for the proposal, including any pertinent documentation.

Interpretations. Upon request, the B89 Committee will render an interpretation of any requirement of the standard. Interpretations can only be rendered in response to a written request sent to the Secretary of the B89 Main Committee.

The request for interpretation should be clear and unambiguous. It is further recommended that the inquirer submit his/her request in the following format:

Subject:	Cite the applicable paragraph number(s) and provide a concise description.
Edition:	Cite the applicable edition of the standard for which the interpretation is being requested.
Question:	Phrase the question as a request for an interpretation of a specific requirement suitable for general understanding and use, not as a request for an approval of a proprietary design or situation.

Requests that are not in this format may be rewritten in the appropriate format by the Committee prior to being answered, which may inadvertently change the intent of the original request.

ASME procedures provide for reconsideration of any interpretation when or if additional information which might affect an interpretation is available. Further, persons aggrieved by an interpretation may appeal to the cognizant ASME committee or subcommittee. ASME does not "approve," "certify," "rate," or "endorse" any item, construction, proprietary device, or activity.

Attending Committee Meetings. The B89 Main Committee regularly holds meetings that are open to the public. Persons wishing to attend any meeting should contact the Secretary of the B89 Main Committee.

METHODS FOR PERFORMANCE EVALUATION OF COORDINATE MEASURING SYSTEM SOFTWARE

1 SCOPE

A critical issue in industrial coordinate metrology is the measurement of a work piece to assure compliance with its dimensional requirements. When using a computerized Coordinate Measurement System (CMS), the usual practice is to correlate computer calculated outputs with the dimensional requirements of the work piece. This correlation is performed by various computer routines that process dimensional coordinate data sets consisting of measurement samples of the object being evaluated.

The purpose of this document is to provide guidelines for evaluating the quality of solutions generated by CMS software and to define minimal documentation requirements for software providers. This Standard is concerned with testing the behavior of algorithm implementation, not the testing of algorithms themselves. It is not the intent of this document to endorse or rate any computational method or system.

Software performance evaluation is useful because it:

(a) allows objective validation of software;

(b) reduces the possibility of error in software application;

(c) defines a method of comparing CMS software.

This document covers the following areas: input data, feature construction, software documentation, performance characterization, and test methodologies.

1.1 Assumptions

The assumptions inherent in this document are as follows.

(a) Measurement uncertainty in coordinate samples is not addressed. For information on measurement uncertainty, refer to Appendix A and ASME B89.3.2, Dimensional Measurement Methods (when released).

(b) Methods to input predetermined samples to the computational system are available.

(c) Personnel have adequate experience and training to implement the evaluation and understand the implications of the results.

1.2 Application

This document is one component required for the evaluation of CMSs. Other relevant documents can be found in Appendix G.

1.3 References

When the following American National Standards referred herein are superseded by a revision approved by the American National Standards Institute, Inc., the revision shall apply.

ASME B89.1.12M-1990 Methods for Performance Evaluation of Coordinate Measuring Machines
ASME B89.4.1-1997, Methods for Performance Evaluation of Coordinate Measuring Machines
ASME Y14.5M-1994, Dimensioning and Tolerancing
ASME Y14.5.1M-1994, Mathematical Definition of Dimensioning and Tolerancing Principles

Publisher: The American Society of Mechanical Engineers (ASME International), Three Park Avenue, New York, NY 10016-5990; ASME Order Department, 22 Law Drive, Box 2900, Fairfield, NJ 07007-2900

References to other documents are for informational purposes only.

2 TERMS AND DEFINITIONS

algorithm: a well defined procedure for solving a particular problem, e.g., sorting algorithms.

Coordinate Measuring System (CMS): any piece of equipment which collects coordinates (points), calculates and displays additional information using the measured points.

datum: a theoretically exact point, line, or plane derived from a feature on a part. See ASME Y14.5M-1994.

datum reference frame (DRF): a part coordinate system constructed from datums. See ASME Y14.5M-1994.

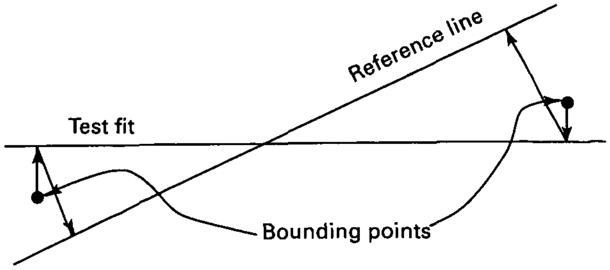

FIG. 1 EXAMPLE OF FIT BOUNDING

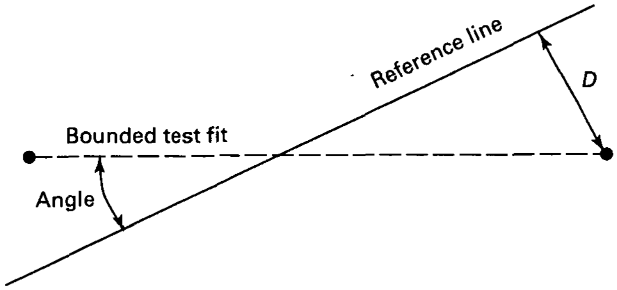

FIG. 2 LINE EVALUATION

least squares fit feature: a feature of perfect orthogonal form, corresponding to a set of data points, which minimizes the sum of the squared deviations between the feature and the individual data points. (Reference Appendix D for additional information.)

objective function: a function which is to be optimized by searching for a minimum (or maximum) as its parameters are varied. A different objective function is used for each type of fit, e.g., least squares versus minimum circumscribed circle.

reference evaluation: the evaluation of the substitute feature using a known implementation of an algorithm.

reference feature: a substitute feature used as the basis for evaluating a test feature.

substitute feature: a feature of perfect geometric form which corresponds to a set of data points and is intended to minimize an objective function.

test: a basic unit of evaluation, based on one or more related data sets, which are applied to one or more software implementations of an algorithm.

test feature: a substitute feature computed by the software under test.

3 SOFTWARE FUNCTIONS

CMS hardware is used to collect data points (raw data) on the surfaces of parts being inspected. CMS software processes these raw data to construct datum reference frames (part coordinate systems) and substitute features that represent the surfaces being inspected. From these constructions, the CMS software can evaluate such characteristics as size, location, orientation, and form.

3.1 Input Data

Raw data to be used to test and analyze CMS software may be obtained by physically inspecting a test workpiece or by mathematical computation. The former represents a test of the entire measuring system, while the latter approach avoids operator, workpiece, environment, and machine influences. The latter approach also makes possible closer control of the raw data sets, including limits on their spatial distribution as well as inclusion of artificially induced form errors. For software analysis, the latter approach is the most universally accepted and the most reliable. This is the approach addressed herein.

3.2 Data Analysis

The raw data points are processed by mathematical algorithms whose purposes are to calculate perfect-form substitute features. First, substitute features are calculated to represent the original data. Then the substitute features are used to evaluate conformance to tolerances or to determine other geometric characteristics of the work piece. An alternative to the use of substitute features is the use of Functional Gage Simulation, described in Appendix F.

Different methods can be used for obtaining substitute features. These methods may have different objective functions, i.e., different criteria for deciding that a particular substitute feature is better or worse than other possible substitute features. Different criteria can, in general, lead to different results. The proper selection of fitting criterion and data analysis method is outside the scope of this document.

Fit criteria are usually based on L_P-norm estimation, or minimum circumscribed, or maximum inscribed methods. Refer to Appendix D for explanations of these methods.

A Datum Reference Frame (DRF) is used to establish the proper relationships of features to their specifications. A discussion of DRFs is contained in Appendix E.

The objective of this Standard is not to decree that any one method is better than any other. Guidance is provided to the user for checking whether particular CMS software produces proper results within the context of the design requirement.

4 PERFORMANCE CHARACTERIZATION

This section establishes the characteristics by which CMS software performance is evaluated. These characteristics are discussed in terms of four categories: quality, robustness, reliability, and ease of use.

Characteristics which are not used for performance evaluation in this Standard are discussed at the end of this section.

4.1 Evaluation of Quality

In this Standard, the quality of the algorithm is evaluated on the basis of the geometric deviation of the test feature from a reference feature.

4.1.1 Evaluation Concept. Some features have unbounded geometry, e.g., lines have infinite length. For the purposes of evaluation, unbounded features are bounded by their sample point sets. The resultant bounded test feature is then compared to the reference feature. Evaluation parameters are defined for each type of feature. See Fig. 1.

4.1.2 Evaluation Parameters. Each feature type has a unique set of evaluation parameters. Test results are reported as outlined below. The figures in this section have the following annotation conventions:

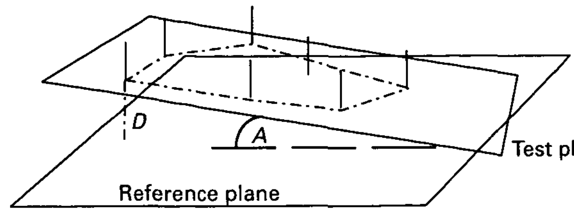

FIG. 4 PLANE EVALUATION

R = reference fit parameter subscript
t = test fit parameter subscript
D = separation distance
r = radius
A = angle
a = cone half-angle

4.1.2.1 Line. The test line is bounded by the perpendicular projection of the sample points onto the test line. The evaluation parameters are (see Fig. 2):

(a) the largest separation distance between the axes of test and reference features;

(b) the angle between the test and reference features.

4.1.2.2 Circle. The test circle is a closed object and naturally bounded. The evaluation parameters are (see Fig. 3):

(a) the absolute value of the difference between the radii of the test and reference circles ($|r_R - r_t|$);

(b) the distance between the centers of the test and reference circles. This may be a three-dimensional distance.

(c) the angle between the planes of the test and reference circles if applicable (see Table 1).

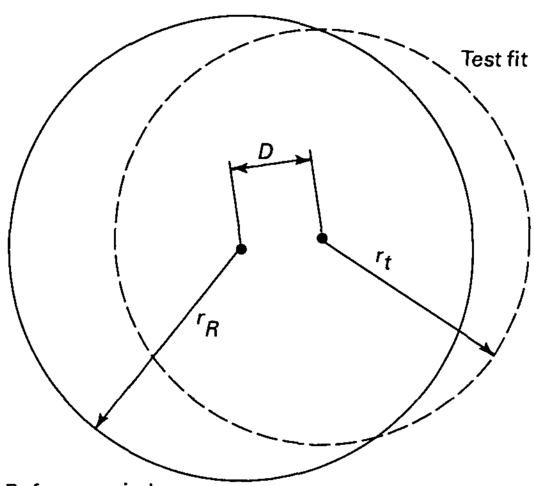

FIG. 3 CIRCLE EVALUATION

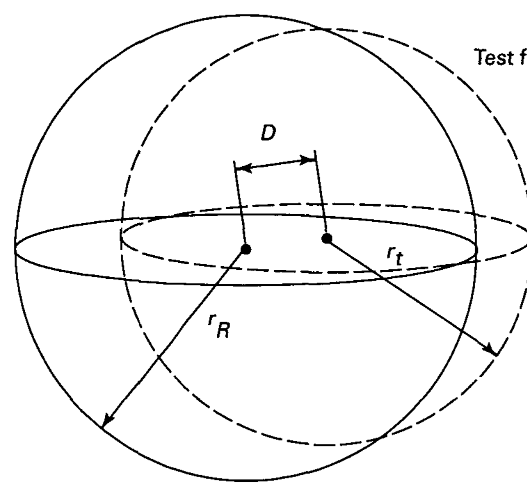

FIG. 5 SPHERE EVALUATION

TABLE 1 CIRCLE FIT TYPES

Circle Fit Type	Reported Angle
Two-dimensional	Not applicable (= 0)
Three-dimensional, both use same ref. plane	Not applicable (= 0)
Three-dimensional, fit plane then two-dimensional circle	Angle between fit planes
Three-dimensional circle fit	Angle between planes

4.1.2.3 Plane. The test plane is unbounded. Sample points are projected onto the test plane for the evaluation. The evaluation parameters are:

(a) the largest perpendicular distance from the reference plane to any projected sample point in the test plane (see *D* in Fig. 4);

(b) the angle between the test and reference planes.

4.1.2.4 Sphere. The test sphere is a closed object and naturally bounded. The evaluation parameters are (see Fig. 5):

(a) the absolute value of the difference between the radii of the test and reference spheres ($|r_R - r_t|$);

(b) the distance between the centers of the test and reference spheres.

4.1.2.5 Cylinder. The test cylinder is bounded along its axis by projecting the sample points perpendicularly onto its axis. It is naturally bounded in circumference. The evaluation parameters are:

(a) the absolute value of the difference between the radii of the test and reference cylinders ($|r_R - r_t|$);

(b) the maximum perpendicular distance from the bounded test cylinder axis to the axis of the reference cylinder (see *D* in Fig. 6);

(c) the angle between the axes of the test and reference cylinders (see *A* in Fig. 6).

4.1.2.6 Cone

(a) The test cone is bounded along its axis by:

(1) projecting the sample data perpendicularly onto the test cone surface; and

(2) projecting these surface points perpendicularly onto the test fit axis (see Fig. 7).

It is naturally bounded in circumference. The reference cone axis is similarly bounded.

(b) The cone evaluation parameters are:

(1) for each cone, the perpendicular distance from the midpoint of the bounded axis to the corresponding cone surface is computed. The evaluation parameter is the absolute difference between these distances ($|r_R - r_t|$);

(2) the maximum perpendicular distance from the bounded test axis to the unbounded reference axis (see *D* in Fig. 8);

(3) the angle between the test and reference axes (see *A* in Fig. 8);

(4) the absolute difference between the test and reference included cone half-angles ($|a_R - a_t|$).

4.1.2.7 Evaluation Parameter Summary. Table 2 summarizes the evaluation parameters for the seven feature geometries dealt with in this Standard. Where three leader dots (...) are used in the table, they indicate that the evaluation parameter for that column does not apply to the geometry for that row.

4.2 Characteristics of Robustness

Robustness is the ability of the software to recover from incorrect inputs, such as colinear data points, too few data points, or, for some CMSs, too many data points. When applicable, robustness shall be tested by including incorrect data sets.

4.3 Characteristics of Reliability

Reliability is the ability of the software to resolve a wide variety of problems. The only reliability characteristic to be addressed is the sensitivity of CMS software to variations of input data. See Appendix A for information about other factors that affect CMS software performance.

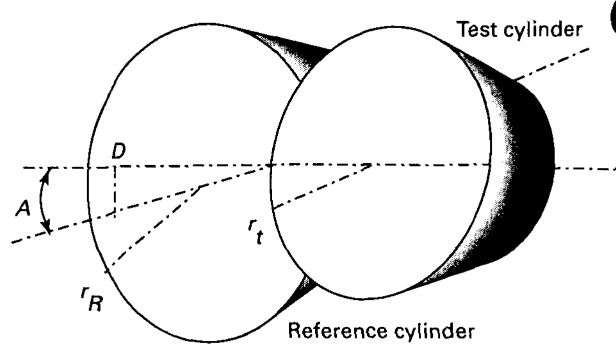

FIG. 6 CYLINDER EVALUATION

TABLE 2 EVALUATION PARAMETERS

Feature	Maximum Distance Between	Angle Between	Radii Difference	Cone Half-Angle Difference
Line	Lines	Lines	...	...
Circle	Centers	Planes	Yes	...
Plane	Projected sample point and reference	Planes	...	...
Sphere	Centers	...	Yes	...
Cylinder	Axes	Axes	Yes	...
Cone	Axes	Axes	At axis centers	Yes

To evaluate CMS software sensitivity, the effects of each factor and interactions among factors should be examined. For each geometric feature type, collections of test data sets shall be designed that include variations in the above factors. See para. 5.6.

4.4 Characteristics of Ease-of-Use

Ease-of-use measures the amount of effort required to use the software, including set-up time, documentation, and structure of the code. This Standard only addresses documentation requirements. Refer to section 6 for more information.

4.5 Related Issues

Software performance may be affected by other factors not included in the performance evaluation. Such factors in the areas of algorithms, computing environment, software implementation, and computational effort are discussed below.

4.5.1 Algorithms. The concept of an *algorithm* is often confused with that of an *implementation* of an algorithm. According to Jackson et al., an algorithm "is a problem solving template that leaves some practical details unspecified. It thus corresponds to a class of computer programs (its implementations) with certain sequences of instructions in each implementation corresponding to the steps of the algorithm." For the purpose of this Standard, algorithms are distinguished by fit criteria (as described in section 3) and by the geometric entity as described in this section. It is important to differentiate between performance comparisons of different implementations of the same algorithm and performance comparisons of different algorithms. Consideration must also be given to the mathematical representation of the problem, i.e., the parameters used, which may have significant effects on the reported results. Strictly speaking, this Standard is concerned with testing the behavior of algorithm implementation, not the testing of algorithms themselves.

4.5.2 Computing Environment. A single implementation of an algorithm may perform differently in various computing environments. The following factors may affect software results:

(*a*) processor characteristics, such as precision and word length
(*b*) computer architecture
(*c*) operating system
(*d*) compiler

4.5.3 Software Implementation. The method of implementing an algorithm may affect its speed and efficiency. Some factors that contribute to this are:

(*a*) programming language
(*b*) use of data structures
(*c*) storage requirements

4.5.4 Computational Effort. The effort or time required to compute the results is affected by the three previous factors. Excessive computing effort can adversely affect throughput of a CMS.

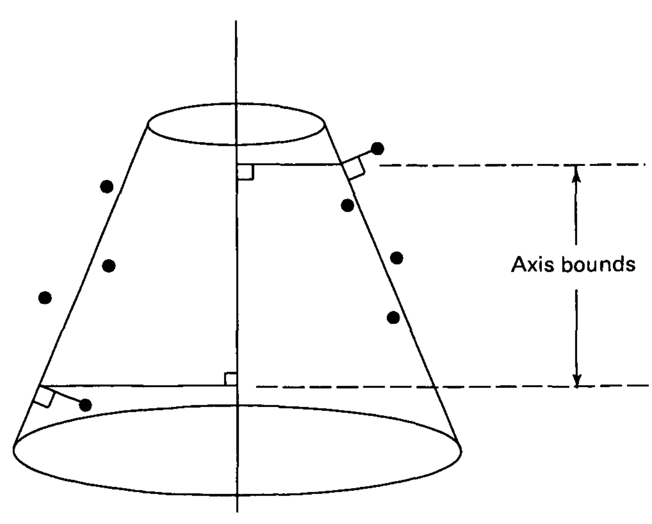

FIG. 7 CONE BOUNDING

5 TEST METHODOLOGIES

This section establishes the general principles, procedures and practices for testing the performance of CMS software.

5.1 Test Principles

For the purposes of this Standard, CMS software is evaluated strictly in terms of its intended function. No assumptions are made regarding the internal structure or operation of the software. The software is subjected to variations of inputs while its outputs are evaluated with respect to a specified objective.

CMS software is tested by the input of sets of test data which reflect the expected range and variability of actual data. Such testing cannot guarantee that the software is completely error free because exhaustive functional testing is impossible.

One implementation of such a software testing system is described in Appendix B.

5.2 Apparatus

The apparatus shall be a testing system interacting with the software under test through exchange of data sets and fit results. Software under test shall be executed in the computing environment in which it will be used. Modifications of the software under test, if any, are limited to those necessary to input the supplied data sets and to extract the fit results. Such modifications shall not change the fit results from what would be produced by the software under test when presented the same data in a production environment.

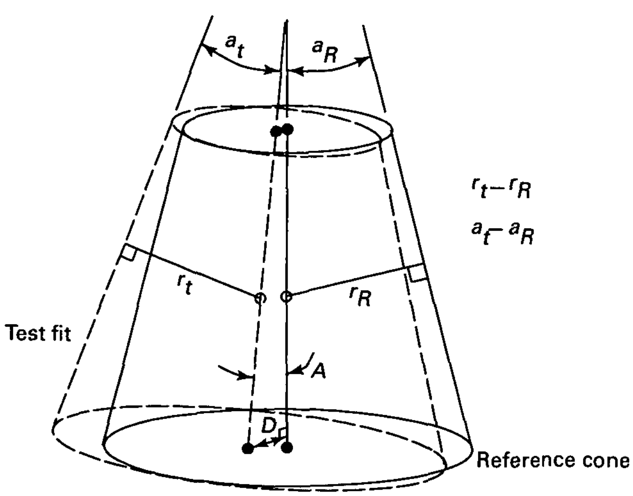

FIG. 8 CONE EVALUATION

The data formats, data resolution, and related characteristics of the test must be defined prior to the test execution.

The person operating the software under test shall be trained in the operation of that software to the extent necessary to input the data, run the software, and gather the output (fit results) in the required format.

Figure 9 presents the three major components of a functional CMS software testing system:

(a) a Reference Pair Generator (RPG) capable of producing reference pairs of data sets and fit results, for specified feature types with controlled range and variability;

(b) a means to transfer data sets to, and receive fit results from, the software under test;

(c) a comparator designed to compare the results of the software under test to the reference results with respect to the objective function and generate an appropriate report.

5.2.1 Reference Pair Generation and Validation

5.2.1.1 Reference Pair Generation. The RPG must be capable of creating a data set and a fit. This may be done in two ways.

(a) A predefined fit result is processed to produce a data set which meets the fit criteria

(b) A data set is generated which approximates the feature and a reference fit is generated from the data by a Reference Algorithm.

5.2.1.2 Validation Reference Pair. In either case, there is a question regarding the validation of the RPG since it is also a complex software program. Because it is not feasible by current technology to prove the correctness of a software implementation of an algorithm, the following is recommended.

(a) The comparator shall evaluate the objective fuction for each test case. The fit yielding the smaller value for the objective function is by definition the better fit.

(b) If the result of the software under test is better than that of the RPG, then that case shall be omitted from the test report. Information sufficient to describe the test case and the reason for omission shall be reported.

(c) The agency responsible for the maintenance of the RPG will be notified of any such omitted cases so that appropriate action can be taken.

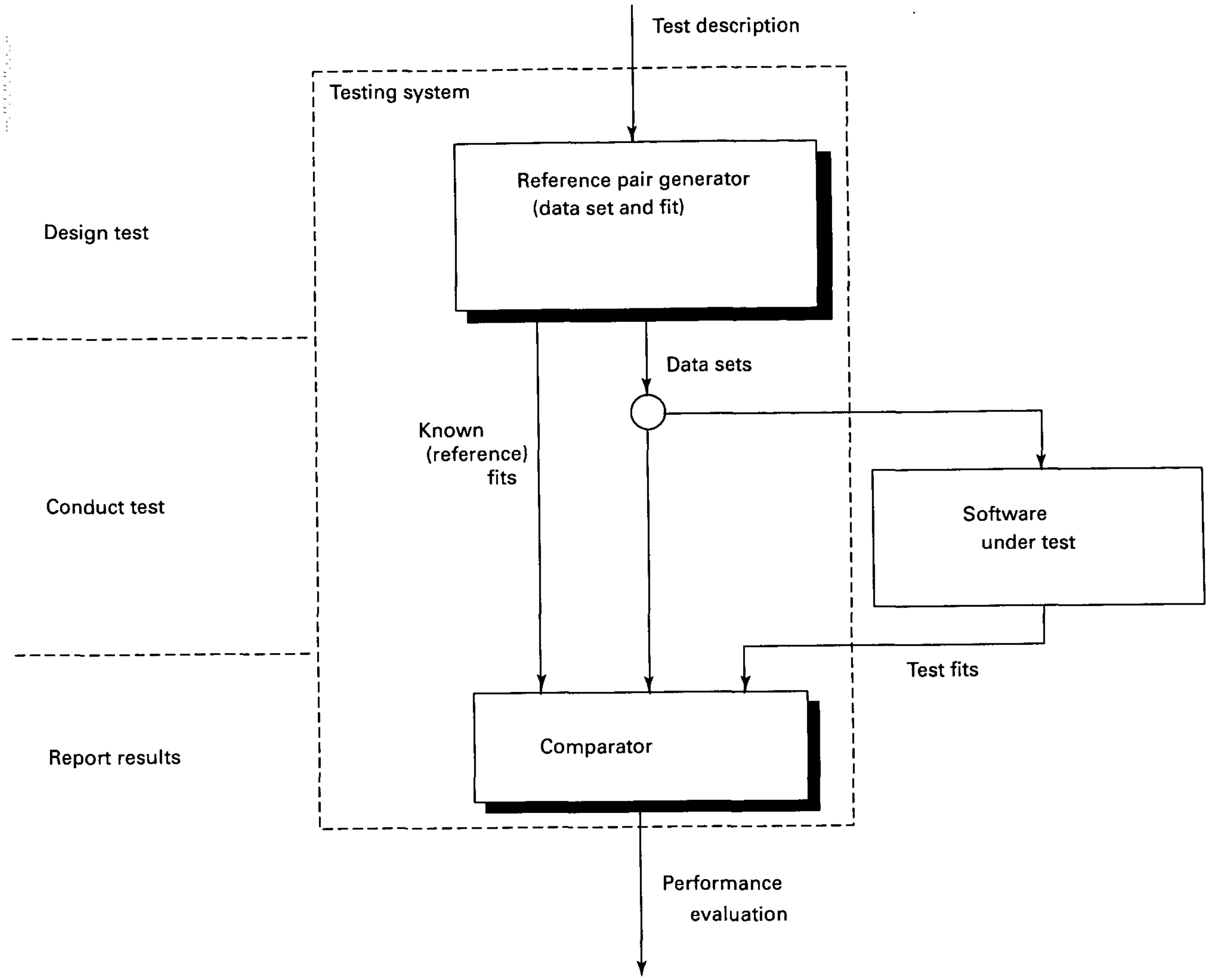

FIG. 9 MAJOR COMPONENTS OF A SOFTWARE TESTING SYSTEM

5.3 Test Procedure

5.3.1 Software Performance Evaluation. CMS software performance evaluation shall include the following steps:

(a) Obtain/generate test data with the testing system.

(b) Process data with software under test.

(c) Collect the outputs of the software under test and reparameterize.

(d) Evaluate the fit results produced by the software under test with the testing system.

5.3.2 Evaluation of Fit Results The following is a generic method for the evaluation of fit results.

(a) The substitute feature is bounded by the sample point set.

(b) A set of parameters for the reference feature is provided.

(c) Evaluation parameters are computed for each pair of test and reference features.

(d) Summary statistics for the evaluation parameters are computed.

5.4 Input Parameters

All input parameters shall appear on the Test Report.

5.4.1 Units. The unit of length shall be agreed upon before the test. The unit of angular measure shall be decimal degrees. All input data and test results shall be reported in the agreed upon units.

5.4.2 Maximum and Minimum Size. The maximum size L_{max} and minimum size L_{min} define the range of feature sizes for test data set generation. The ratio L_{max}/L_{min} shall be no greater than 10^4. Greater ranges can be accounted for using more than one test, each satisfying this range requirement.

5.4.3 Farthest Position. The farthest distance from the origin that a feature can be placed shall be specified and indicated in the test report. This distance shall be at least $2L_{max}$.

5.4.4 Types of Features. The input parameters will define the types of features to be evaluated from the set of features supported by this Standard. Only features supported by the CMS may be evaluated.

5.4.5 Maximum Number of Sample Points. The maximum number of sample points used for test generation shall be agreed upon before the test and shall be indicated on the Test Report.

5.4.6 Test Data Precision. The precision to which the test data is generated shall be agreed upon before the test but shall be at least as precise as $10^{-5}L_{min}$. This does not and should not restrict the precision to which fits are computed.

5.4.7 Seed Values. CMS software may require seed values. These values are typically defined by the first few sample points, i.e., a cone seed may require 3 points for a smaller circle followed by 3 points for a larger circle. If the test data is constructed to provide such seed values, it shall be noted on the test report for each feature type.

5.4.8 The Default Test. The default test is defined by the following default input parameters: units in millimeters; $L_{min} = 1$, $L_{max} = 500$, farthest position = 1000, maximum number of sampled points = 500. The test data sets shall be generated to a precision of 10^{-5}.

5.5 Generation of Test Data

For each feature type, 30 data sets shall be generated via computer simulation satisfying the following requirements for size, position, orientation, number of sampled points, sampling plans, and form errors.

5.5.1 Sizes. The sizes of features shall are bounded by L_{min} and L_{max}. The 30 sizes shall be determined within three size categories as follows: Generate ten random numbers in each range of (0, $\frac{1}{3}$), ($\frac{1}{3}$, $\frac{2}{3}$), and ($\frac{2}{3}$, 1). For each random number x, define the size of the feature as $L_{min}^{1-x}L_{max}^{x}$. The size parameters for the feature types are defined as follows:

Line	Bounded length
Plane	Maximum of length and width of the bounding rectangle
Circle	Diameter
Sphere	Diameter
Cylinder	Maximum of diameter and bounded height
Cone	Maximum of base diameter and bounded height

For example, when generating lines with $L_{min} = 1$ and $L_{max} = 1000$, three size scales would be created by using the above generation scheme. Ten line segments would have sizes between 1 and 10 units, ten would have sizes between 10 and 100 units, and ten would have sizes between 100 and 1000 units.

Lines and circles can be tested in two- or three-dimensions. A two-dimensional line or circle is restricted to be parallel to a coordinate plane.

The aspect ratio of planes, the height-to-diameter ratio for cylinders, and the height-to-base-diameter ratio for cones shall be between 0.02 and 10. Specifically, in each size category, one ratio shall be 0.02, two shall be 0.2, four shall be 1, two shall be 3, and one shall be 10. For each size category the degree measures of the apex half-angles ψ for cones shall fall into the ten ranges defined by the eleven values, $\frac{1}{2}$, 1, 2, 4, 10, 20, 30, 40, 60, 80, and 85.

5.5.2 Positions. Test cases shall include some features for which the data points have their centers of mass near the origin and some for which they are far from the origin, close to the specified farthest position. Feature positions are not restricted to the first quadrant or octant unless such is a special input restriction for the software under test, in which case, the restriction shall be noted on the test report.

5.5.3 Orientations. In each size category, the test cases shall include nominal orientations which are aligned with each of the coordinate axes of the data and one aligned with a vector whose direction is (1, 1, 1). Except for these, orientations shall be determined randomly.

5.5.4 Numbers of Points. For each feature type, one data set in the middle size category shall be comprised of the minimum number of points shown below. Also, one data set shall be comprised of the

TABLE 3 NUMBER OF REQUIRED FORM ERRORS

	Line	Plane	Circle	Sphere	Cylinder/Cone
One-dimensional sine [Note (1)]	4 (0.5, 1, 2, 3)	...	5 (1, 2, 3, 4, 5)	...	1 (0.5) axis sine
Surface sine [Note (1)]	...	4 (0.5, 0) (0.5, 1) (1, 1) (3, 1)	...	6 (0, 0.5) (0, 1) (2, 1) (3, 1) (2, 0.5) (3, 2)	5 (0, 0.5) bow (0, 0.5) hourglass (2, 0) 2-lobed (3, 0) 3-lobed (3, 1) combination
Step	1	1	2	...	1 (radial step about axis)
Bend	2	2	...	...	...
Taper	...	...	...	...	1
Random	2	2	2	3	1
None	1	1	1	1	1

NOTE:
(1) Frequencies in parentheses.

specified maximum number of points. The remaining data sets shall be comprised of numbers of points strictly between these minimum and maximum values, chosen using a logarithmically random generator. For surfaces these randomly generated numbers may be rounded off to a convenient composite number suitable for a grid pattern, provided the number of points is still strictly between the minimum and maximum values. The minimum numbers of points are:

Line	2	Sphere	4
Plane	4	Cylinder	6
Circle	3	Cone	6

5.5.5 Sampling Plans

5.5.5.1 Distribution. The points in each data set shall be nominally regularly spaced. In the cases of cylinders and cones, some distributions lead to multiple solutions. Two parallel rings of three points each can yield two correct, orthogonal fits. Eight points distributed on the corners of a box yield three correct, orthogonal fits. Care must be taken to avoid distributions that are close to these ambiguous cases. This may be ignored when seed values are used to establish approximate orientation.

5.5.5.2 Partially Sampled Surfaces. Surfaces may be partially sampled, representing cases where the entire feature is not accessible or incomplete, e.g., bearing face or surface patch of a taper.

Sampled arcs of circles, cones, and cylinders shall be 90 deg, 180 deg, and 360 deg. In each of the three size categories, two data sets shall represent 90 deg samples, two 180 deg samples, and the remaining six 360 deg samples.

Spheres shall be sampled over 90 deg and 180 deg polar patches and an equatorial band defined by a ±15 deg angle (30 deg total) from the center. In each of the three size categories, three data sets shall be sampled over 90 deg polar patches, five over 180 deg polar patches, and two over equatorial bands.

5.5.6 Form Errors. One data set in each size category shall have no form error (the one in the middle size category using the minimum number of points). When applicable, these three data sets shall not coincide with extreme values of aspect or height-to-diameter ratios or extreme values of a cone's apex half-angle. The remaining test cases shall include a maximum peak-to-peak form error of either 0.1% (four test cases per size category) or 2% (five test cases per size category) of the feature's length scale. The length scales for the feature types are defined as follows:

Line	Bounded length
Plane	Minimum of length and width of the bounding rectangle
Circle	Diameter
Sphere	Diameter
Cylinder	Minimum of diameter and bounded height
Cone	Minimum of base diameter and bounded height

The number and type of required form errors for each size category for each feature type are identified in Table 3; their mathematical definitions are given in Appendix I. Each form error identified shall coincide with a 2% form error at least once.

In addition to these errors, uniform random errors shall be superimposed as follows: If the maximum peak-to-peak error was 2% of the feature's length scale, then a three-dimensional, uniformily random error of size 0.1% of the feature's length scale shall be added. If the maximum peak-to-peak error was 0.1% of the feature's length scale, then a three-dimensional, uniformily random error of size 0.01% of the feature's length scale shall be added. Lines and circles can be tested as two- or three-dimensional features. When testing as a two-dimensional feature, these random errors shall be restricted to the plane of the feature.

5.6 Test Set

At a minimum, data sets described in para. 5.5 shall be generated. Additional tests may be run to uncover specific problems if required. The guidelines for test generation shall be followed except where they violate stated CMS vendor specifications. Where such exceptions occur, they shall be noted on the test report.

The suggested feature parameterization of Appendix B is used to define the nominal feature. From the nominal feature, a sample set is generated using the guidelines in para. 5.5.

5.7 Process Data With Test Software

Special conversion software or a modified version of the CMS software may be required to allow for the introduction of data not acquired through the normal CMS data input channel. If the CMS is capable of executing a stored program, a program which performs data set evaluations must be written. This program may be subsequently used to evaluate new versions of CMS software. CMS systems without stored program capability may be manually controlled to perform their evaluations, but it is recommended that automatic methods be made available if possible.

The order of the points in the data set may be changed to satisfy any special requirements of the software under test. If reordering of the data is required, it shall be noted on the test report.

The results of the algorithms should be output in a format compatible with the comparator function. For a suggested format refer to Appendix B.

5.8 Calculation and Interpretation of Results

The guidelines for algorithm comparison in para. 4.1 shall be used to compare the results of the software under test to the reference results for each data set. For each geometric feature type, a statistical analysis shall be performed to evaluate the root-mean-square (RMS) and maximum magnitude of the observed evaluation parameter values.

If unambiguous, objective criteria are used to identify so-called "bad fits," such fits may be excluded from the statistical analysis and reported separately. Difference angles are to be expressed in microradians. Distance and radii differences shall be converted to the normal units of the CMS (see para. 5.4.1).

5.9 Reporting of Test Results

A test report shall be produced at the conclusion of the comparison phase. The test report shall include the following information:

(a) the reference software used and its version identifier;

(b) the characteristics of the software that was tested (including computing environment, software version, and any other necessary identifying characteristics);

(c) the geometric feature types tested;

(d) any reordering of the data or seed values;

(e) the range of conditions represented by the test data for each geometric feature type;

(f) the RMS value of each evaluation parameter for each geometric feature type;

(g) the maximum observed value of each evaluation parameter for each geometric feature type;

(h) the criteria for identifying bad fits for exclusion from the statistical analysis;

(i) the test results for bad fits excluded from the statistical analysis and the corresponding test data characteristics.

If no fits were excluded from the analysis, the RMS statistic includes the effects of both systematic and random deviations between the software under test and the reference results. Thus, it can be interpreted as the expected deviation from true value for the software under test, over the range of conditions represented by the test data. To support this interpretation, the effects of uncertainty inherent in the reference results must be included in an uncertainty statement for the RMS statistic.

If any test results were excluded from the analysis, the above interpretation of the results does not hold. Rather, the software is unreliable for the conditions of the test. Although there is no consistent metrology interpretation of the test results in this case, the results have diagnostic value.

If the default test is used, the following minimum values shall be used where applicable in the Test Report when reporting RMS or maximum observed values for evaluation parameters:

Distances	10^{-5} μm
Angles	10^{-7} arc sec

In the case the RMS or maximum value of an evaluation parameter is below the minimum, the reported value shall be reported as "less than 10^{-5}" or "less than 10^{-7}," as appropriate, along with the corresponding units.

5.10 Periodic Reverification

CMS software should be evaluated when upgraded software is released, when there is any change in the computing environment that might affect the results, or when results reported by the software appear to be abnormal.

6 SOFTWARE DOCUMENTATION

This section provides guidelines for minimum documentation for coordinate metrology software.

6.1 Purpose

The purpose is to provide guidelines for preparation of user documentation by CMS manufacturers that will provide, to the software users, a sufficient understanding of the intent and underlying principles of each software procedure used in the analysis of coordinate data.

6.2 Compliance

Coordinate metrology software meet the minimum documentation requirements of this Standard if the guidelines listed under para. 6.3 are followed. The guidelines are for content only, not for format or structure. These guidelines apply to each procedure, or set of procedures, that are applied to a specific dimensioning and tolerancing call-out. For an example reference Appendix C.

6.3 Required Information

The information listed below is the minimum required for proper documentation.

6.3.1 Procedure Name. A name used to designate each algorithm implementation.

6.3.2 Brief Description. A one-line description of the procedure.

6.3.3 Standards Compliance. Compliance with applicable gaging standards should be included in this section.

6.3.4 Explanation of Procedure. A detailed description of the procedure and services should be provided as follows.

6.3.4.1 Intent. A concise discussion of the intent of the procedure(s) should be provided. This discussion should address all aspects of the procedure, including input data, calculations and other data processing, and method of part evaluation with respect to tolerance requirements.

6.3.4.2 Underlying Principles. Any underlying principles that the user may need to understand in order to properly use the procedure (to the extent that such knowledge may not be assumed for a skilled operator in general).

6.3.4.3 Illustrated Examples. An illustrated example that describes the relationship of output to input should be provided for each procedure that applies to a specific dimensioning and tolerancing evaluation. This example should graphically show sufficient data points, tolerance zone, and the results in relation to the tolerance zone. If a numerical result is calculated and compared to the tolerance zone, then this result should be graphically displayed. Any applicable datum feature(s) and their relationship(s) to the tolerance zone should be illustrated. The illustration must clearly show the intent of the procedures in relation to the individual data points.

6.3.4.4 Limitations and Precautions. Limitations of the procedure and other precautions to the user.

6.3.5 Input. Descriptions, formats, and examples of the access to the procedure(s) should be provided.

6.3.5.1 Defaults. Default input(s) should be provided.

6.3.5.2 Required Inputs. A description of all required input(s) to the procedure(s) should be provided.

6.3.5.3 Optional Inputs. All optional input(s) should be defined.

6.3.5.4 Interface Equivalence. Description, formats, and examples of all equivalent input statement(s) should be listed for any supported interface.

6.3.5.5 Input Limitations. Known limitations and constraints on the procedure(s) input should be listed, e.g., minimum and maximum number of coordinate points the procedure can process.

6.3.6 Output. Descriptions, formats, and examples of the outputs of the procedure(s) should be provided.

6.3.6.1 Defaults. Descriptions, formats, and examples of default output(s) should be provided.

6.3.6.2 Optional Outputs. Descriptions, formats, and examples of optional output(s) should be provided.

6.3.6.3 Interface Equivalence. Equivalent output statement(s), format(s), and example(s) for any supported interface should be listed.

6.3.6.4 Output Limitations Known limitations of output should be provided, e.g., precision.

6.3.7 Exception Conditions. Listing and definition of the various exception handling procedures should be provided.

6.3.8 Computational Uncertainty. A number characterizing the expected error contributed by the software. The number should include the estimated cumulative effects of all computational factors that affect geometric uncertainty, including numerical rounding, convergence criteria used in estimation algorithms, and other factors independent of specific measurement tasks. This number should be a typical value, with the understanding that some applications may have errors that exceed the stated quantity. Reference para. 4.5 for information on related issues.

6.3.9 Associated Datum Features. Reference to datum features documentation (if applicable) should be provided.

MANDATORY APPENDIX I
MATHEMATICAL DESCRIPTIONS OF FORM ERRORS

To describe the form errors, a perfect, nominal feature is first described, having a convenient location and orientation. The form errors are then described in this position, as well as a description of the form error. These features would be translated and rotated in the actual test.

(a) Nominal Features

(1) Line. A line segment having endpoints (0, 0, 0) and $(L, 0, 0)$.

(2) Plane. A rectangle having corners (0, 0, 0), $(L, 0, 0)$, and $(0, W, 0)$.

(3) Circle. A circle in the x-y plane centered at the origin, defined in polar coordinates by $r = R$.

(4) Sphere. A sphere centered at the origin, defined in spherical coordinates by $\rho = R$.

(5) Cylinder. A truncated cylinder defined in cylindrical coordinates by $r = R$ and having extent from $z = 0$ to $z = h$, where h is the height of the cylinder.

(6) Cone. A frustum defined in cylindrical coordinates by $r = R + z\sin\psi$ and having extent from $z = 0$ to $z = h$, where h is the cone's height and ψ is the cone's apex angle.

Let A denote the desired amplitude of the error.

(b) 1-D Sine Errors of Frequency v

(1) Line. $z = A\sin(2\pi xv/L)$.

(2) Circle. $r = R + A\sin(v\theta)$ expressed in polar coordinates.

(3) Cylinder and Cone. Points are shifted from the nominal in the x-direction by an amount $A\sin(2\pi xv/L)$.

(c) Surface Sine Errors of Frequencies v_1, v_2

(1) Plane. $z = A/2[\sin(2\pi xv_1/L) + \sin(2\pi yv_2/W)]$.

(2) Sphere. $\rho = R + A/2[\sin(v_1\theta) + \sin(v_2\psi)]$ expressed in spherical coordinates.

(3) Cylinder. $r = R + A/2[\sin(v_1\theta) + \sin(2\pi zv_2/h)]$ expressed in cylindrical coordinates.

(4) Cone. $r = R + z\sin\psi + A/2[\sin(v_1\theta) + \sin(2\pi zv_2/h)]$ expressed in cylindrical coordinates.

For the "hourglass" form error for cylinders and cones, replace $2\pi zv_2/h$ with $(\pi + 2\pi zv_2/h)$ in the preceding two equations. (v_2 would be 0.5 in these cases.)

If $v_1 = 0$ or $v_2 = 0$, replace $A/2$ with A in the above equations.

(d) Step Errors

(1) Line. If $x > x^*$, $z = A$, else $z = 0$; x^* is chosen randomly between $L/4$ and $3L/4$.

(2) Plane. If $ax + by + c > 0$, then $z = A$, where $ax + by + c = 0$ defines a line (in the x-y plane) chosen randomly but passing through the rectangle having corners $(L/4, W/4, 0)$, $(3L/4, W/4, 0)$, and $(L/4, 3W/4, 0)$.

(3) Circle. If $0 \leq \theta \leq \theta^*$, then $r = R + A$, where θ^* is chosen randomly between 90 deg and 180 deg.

(4) Cylinder. If $0 \leq \theta \leq \theta^*$, then $r = R + A$, where θ^* is chosen randomly between 90 deg and 180 deg.

(5) Cone. If $0 \leq \theta \leq \theta^*$, then $r = R + z\sin\psi + A$, where θ^* is chosen randomly between 90 deg and 180 deg.

(e) Bend Errors of Angle α

(1) Line. If $x > x^*$, then $z = (x - x^*)\tan\alpha$, else, $z = 0$; x^* is chosen randomly between $L/4$ and $3L/4$.

(2) Plane. If $ax + by + c > 0$, then $z = (ax + by + c)^*\tan\alpha$, where $ax + by + c = 0$ defines a line (in the x-y plane) chosen randomly but passing through the rectangle having corners $(L/4, W/4, 0)$, $(3L/4, W/4, 0)$, and $(L/4, 3W/4, 0)$.

(f) Taper of Angle α

(1) Cylinder. If $z > z^*$, then $r = R + (z - z^*)\tan\alpha$; else $r = R$, where z^* is chosen randomly between $h/4$ and $3h/4$.

(2) Cone. If $z > z^*$, then $r = R + z\sin\psi + (z - z^*)\tan\alpha$; else $r = R + z\sin\psi$, where z^* is chosen randomly between $h/4$ and $3h/4$.

NONMANDATORY APPENDIX A
FACTORS THAT INFLUENCE THE RESULTS

A1 FACTORS OF SOFTWARE AND COMPUTATIONAL ENVIRONMENT

The following factors affect the quality of computations carried out by CMS software.

(a) *Feature Geometry.* CMS software behavior may be affected by a feature's geometry, notably its size and location. Depending upon data manipulation techniques employed, software may be less reliable for features of large size or features located far from the origin.

(b) *Feature Form Error.* Errors of form (straightness, roundness, cylindricity, etc.) of measured features affect the calculations of position, size, and orientation by software. Strong interactions between form error and sampling strategy are likely.

(c) *Feature Sampling Strategy.* The number of sampled points and the pattern in which those points were taken may affect CMS software reliability. In most cases, the mathematical minimum number of points necessary to determine a geometric element is not sufficient for the measurement of an actual feature. Strategies of point density and pattern sampling can be found in BS 7172-1989.

(d) *Point Measurement Error.* Errors in each sampled point, that were induced by the point measurement process, may affect the reliability of CMS software. However, this issue is beyond the scope of this Standard; see the ISO Guide to the Expression of Uncertainty in Measurement for information about the propagation of errors through calculations.

A2 FACTORS OF IMPLEMENTATION

The output accuracy of a CMS is also influenced by a combination of factors beyond the influences of software and the computational environment. The CMS user should be aware of these factors and make every effort to control their influence. These factors include:

(a) the accuracy characteristics of the coordinate data, as determined by proper verification. In the case of coordinate measuring machines, they should be verified per ASME B89.1.12M.

(b) the physical environmental effects on the CMS and the workpiece;

(c) the effects of the use of substitute geometry by the CMS software and the resulting uncertainty when measuring geometric features;

(d) the factors that affect the sensitivity and behavior of the algorithms, including:

(1) point measurement errors on imperfect surfaces caused by less than the minimum number of points (point density) needed to identify a feature;

(2) sampling errors on imperfect surfaces resulting from poor placement or inadequate coverage of the characteristic being sampled;

(3) workpiece form or positional errors caused by improper measurements and the variables introduced by the mathematics.

A3 FACTORS OF ALGORITHM SELECTION

Software algorithms, like any other tools of manufacturing, may be misused or misapplied. Factors that must be considered in the selection of software for a measurement task include the following:

(a) the choice of the objective function to evaluate a geometric requirement;

(b) the use of two-dimensional software to inspect a three-dimensional characteristic does not necessarily allow for required degrees of freedom, e.g., MMC positional tolerances;

(c) the CMS part program may not meet the geometric requirements of the workpiece as expressed on the engineering drawing.

NONMANDATORY APPENDIX B
NIST ALGORITHM TESTING SYSTEM (ATS)

B1 INTRODUCTION

The National Institute of Standards and Technology (NIST) has developed a software system that determines the performance characterization according to section 4. This Appendix describes the operation of this system, called the NIST Algorithm Testing System, or ATS.[1] Section B2 summarizes the operation of the ATS. Section B3 describes the file formats used to exchange data between the ATS and software under test.

B2 ATS OVERVIEW

The ATS is a software package that evaluates the performance of fitting software. The ATS consists of three components: a data generator, a set of reference algorithms, and a fit analysis module. The data generator produces data sets based on a test description provided by the ATS operator. In this test description, the ATS operator specifies:

(a) nominal (ideal) geometry of the feature;

(b) form errors of the feature being simulated;

(c) sampling plan (distribution of points on the feature);

(d) (random) measurement error distribution for the points.

The data sets are processed by the CMS software to generate Test Fits and by the reference algorithms to generate Reference Fits. The test and reference fits are compared to assess performance of the CMS software.

B3 FILE FORMATS

Two classes of data are exchanged between the ATS and the software unit under test: data sets of 3-dimensional coordinate values generated by the ATS, and the results of fits to these data sets generated by the software under test. All file formats are ASCII text representations of the data.

B3.1 Data Set Files

The ATS generates a separate file for each data set sent to the software unit under test. The data set files produced by the ATS have the following format:

```
Line 1:  ID            GEOMETRY_TYPE DATE
Line 2:  COUNT
Line 3:  COORDINATES
Line 4:  COORDINATES
         ...
```

In Line 1, ID is an identifier string, of no more than 32 characters, that uniquely identifies the data set within the ATS. GEOMETRY_TYPE is a label of the geometry type represented by the data. The geometry types currently supported in the ATS are: circle, cone, cylinder, line, plane, sphere, and torus.

Continuing on Line 1, DATE is the date on which the data file was generated, represented in the format: dd-mmm-yyyy (e.g., 12-JUL-1993). On Line 2, COUNT is an integer count of the number of points represented in the rest of the file. Starting on Line 3, each remaining line represents one point of the data set, using three Cartesian coordinate values in order *X, Y,* and *Z.* The coordinate values are dimensionless, but are all to the same scale. Coordinate values are represented in decimal floating point notation, separated by spaces. Exponential (scientific) notation is not used. The coordinate values are represented with as many digits as necessary to represent the values to the resolution used in generating the data. Trailing zeroes are not represented. Each line is terminated by an end-of-line sequence (ASCII codes 13 and 10) following the *Z* coordinate value. After the end-of-line sequence for the last point, the file ends.

B3.2 Fit Result Files

Fit results are returned to the ATS in files formatted according to the geometry type. Each file represents a single fit result for a single data set. The first line of the fit result file should be a copy of the first line of

[1] Information on the availability and use of the ATS can be obtained directly from NIST. The name Algorithm Testing System is slightly misleading, since the system is designed to test software that implements a fitting algorithm, not the mathematical algorithm itself.

TABLE B1 SEQUENCE FOR REPORTING FIT PARAMETERS

	Line	Plane	Circle	Sphere	Cylinder	Cone	Torus
Position							
X	1	1	1	1	1	1	1
Y	2	2	2	2	2	2	2
Z	3	3	3	3	3	3	3
Orientation							
X	4	4	4	...	4	4	4
Y	5	5	5	...	5	5	5
Z	6	6	6	...	6	6	6
Diameter							
Major	...	...	7	4	7	...	7
Minor	...	...	...	...	...	...	8
Axis Location							
	...	...	...	...	...	7	...
Half-Angle							
	...	...	...	...	...	8	...

GENERAL NOTES:
(a) The position parameter for lines is any point on the line; for cylinders and cones it is any point on the axis; for planes it is any point on the plane.
(b) The orientation parameters must be proportional to the direction cosines for the indicated direction. For planes, the orientation is that of the plane normal.
(c) The axis location parameter for a cone is the perpendicular distance from the position point to the surface of the cone.

the data set file from which the fit was calculated. The remainder of the file contains the parameters of the fit. The parameters and their sequence expected in the file are described below. Parameter values must be separated by white space. (White space consists of the ASCII codes for space, tab, carriage return, line feed, or form feed.) Any line after the first may contain comments. A comment starts with a semicolon (;) and continues to the end of the line. Comments are treated like white space. All position and size parameters are dimensionless, but must be reported in the same scale as the data set coordinates. All angle values must be in decimal degrees. Parameters may appear in decimal floating point or exponential (i.e., ±x.xxe±xx) notation. The precision of the parameters is assumed to be exact; that is, the values are assumed to include trailing zeroes to infinity.[2]

The fit parameters expected in result files correspond to the parameterization used in the ATS to calculate the difference between fits as described in section 4. These parameters and their expected sequence in the file are shown in Table B1.

[2] Since fit comparisons involve calculation of geometric differences, rather than direct comparison of parameter values, fit parameters should be expressed in as much precision as possible. In particular, it is important to *not* round the fit parameters to the apparent resolution of the data set.

NONMANDATORY APPENDIX C
EXAMPLE DOCUMENTATION

This Appendix presents an example of acceptable documentation. The example is *not* necessarily acceptable measurement practice.

DISCLAIMER: The sole purpose of this example is to demonstrate adequate documentation practice and should not be construed as explicitly or implicitly endorsing or requiring any single method of calculation, input, output, illustration, etc. A hypothetical brand CMM, XCMM with a native language XMML is used in the following example.

In this example, 15 points have been measured on a surface and assigned to a set called PLANE1 and are to be evaluated against a tolerance of 0.010 in.

C1 PROCEDURE NAME

The procedure name is *flatness.*

C2 BRIEF DESCRIPTION

This procedure calculates the flatness of a plane.

C3 STANDARDS COMPLIANCE

Calculations of flatness comply with the following standards: Standard XXX and Standard YYY.

C4 EXPLANATION OF PROCEDURE

To calculate the flatness of a geometric plane, using data points which are a sample of the surface which approximates the plane and then evaluate it against a tolerance value.

C4.1 Intent

A least squares plane is calculated from the measured points assigned to the set PLANE1. The distances between the least squares plane and the two extreme points on each side of this plane is calculated, e.g., 0.0011 on one side and 0.0022 on the other. These distances are added with the result being the calculated flatness value, e.g., 0.0033. This calculated difference is compared to the tolerance (0.010 − 0.0033).

C4.2 Underlying Principles

To find an ideal plane, the sum of the squares of the normal distances from each point to the plane is a minimum. Once this plane is determined, the farthest point on each side of the plane is resolved. The distance between these two points is calculated, normal to the plane, and identified as the flatness.

C4.3 Illustrated Example

See Fig. C1.

C4.4 Limitations and Precautions

Flatness procedure can be accessed in three ways:

(*a*) pressing the = symbol on the keypad and typing in the name PLANE1. At the prompt, enter the tolerance value of 0.010;

(*b*) type in the XMML command:

```
fltns (ele=PLANE1,tol=0.010)
```

(*c*) through the FORTRAN statement:

```
CALL FLTNS('PLANE1',0.010)
```

C5 INPUT

C5.1 Defaults

If no tolerance value is entered, the procedure will default to 0.025 in.

C5.2 Required Inputs

The name of the set of points (PLANE1 in this case) must be input.

C5.3 Optional Inputs

A statistics terminal display option is available through the XMML command by adding "sta=term." The resulting command would be:

```
fltns (ele=PLANE1,tol=0.010,sta=term)
```

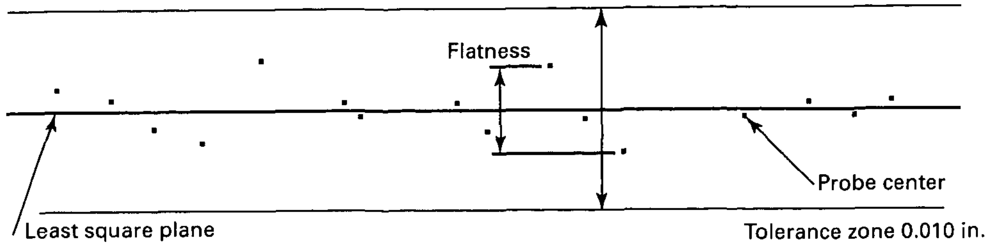

FIG. C1 FLATNESS EXAMPLE

C5.4 Interface Equivalence

The DMIS interface equivalent statements are:

T(FLAT) = TOL/FLAT,0.010
EVAL/F(PLANE1),TA(FLAT)

C5.5 Input Limitations

The maximum number of points that can be computed is 99. The minimum number of points is 6.

C6 OUTPUT

The flatness value is printed in the following default format:

FLATNS of: $$$$$$$ = ##.#### in.......###.#% of #.#### TOL.

If the calculated value is greater than the tolerance, the characters OUTOFTOL are printed on the next line.

In this case, the calculated flatness is 0.0033 and the output would read:

FLATNS of: PLANE1 = 0.0033 in....... 33.3% of 0.0100 TOL.

C6.1 Defaults

The above is the default format.

C6.2 Optional Output

An additional optional output format is the statistics. If this option is exercised, a histogram of the individual point deviations are displayed on the terminal but are not printed.

C6.3 Interface Equivalence

The DMIS interface equivalent output statement is:

TA(FLAT) = TOL/FLAT,0.010, INTOL

C6.4 Output Limitations

The output limits are 4 decimal places (inches) or 3 decimal places (metric).

C7 EXCEPTION CONDITIONS

The CMS system outputs the following error messages when exception conditions occur.

SPATIAL DISTRIBUTION ERROR means that the points are outside the prescribed distribution, indicating that one or both of the following rules were violated.

(a) The thickness must be less than half the width.

(b) the width must be greater than one-tenth the length.

Either remeasure surface taking care not to exceed these rules, or delete points outside of this spatial boundary and recalculate.

POINT NUMBER MAX means that over 99 points have been submitted to the procedure for calculation. Remeasure surface taking 99 or less points or delete points until 99 remain and recalculate.

POINT NUMBER MIN means that less than 6 points have been submitted to the procedure for calculation. Remeasure surface taking at least 6 points.

C8 COMPUTATIONAL UNCERTAINTY

The least squares fitting software was evaluated in accordance ASME B89.4.10 and found to have an RMS deviation of 10^{-8} m for plane separation and 0.02 arc sec for plane tilt.

C9 ASSOCIATED DATUM FEATURES

Flatness is not computed with respect to any other features.

NONMANDATORY APPENDIX D SUBSTITUTE FEATURES

This Appendix is directed at the computer programmer concerned with developing substitute feature software.

A substitute feature is a perfect-form geometry (circle, plane, cylinder, etc.) used to represent an actual feature during subsequent part evaluation. A substitute feature is the "representation" of the measured data points. This Appendix describes the most common methods used to define the substitute feature.

Fit criteria lead to an optimization problem, the solution of which defines the parameters of the substitute geometry. With some exceptions, more than one substitute feature may optimize any one criterion. Any application sensitive to such ambiguities must guard against them to assure proper results. A discussion of fitting problems is beyond the scope of this Appendix.

The mathematical model used in this Appendix is a substitute feature characterized by a vector of parameters $\boldsymbol{b}$. The perfect-form geometry is defined by a function $f_b(\boldsymbol{p})$ that assigns a real number to every point $\boldsymbol{p}$ in space. The substitute feature surfaces is described by the equation $f_b(\boldsymbol{p})=0$. The entire space is divided into two half spaces by the inequalities $f_b(\boldsymbol{p}) < 0$ and $f_b(\boldsymbol{p}) > 0$. Any particular geometric form can be represented by a wide range of functions $\mathbf{f}$. In this Appendix, the only restrictions on the functional form of $\mathbf{f}$ are features of size (i.e., circles, cylinders, spheres, parallel lines, and parallel planes), the half space $f_b(\boldsymbol{p}) < 0$ correspond to the intuitive notion of "inside the feature," and the half space $f_b(\boldsymbol{p}) > 0$ correspond to the "outside" of the feature. A particular functional form $\mathbf{f}_b$ may involve constraints on $\boldsymbol{b}$ to maintain the validity of the representation. Such constraints are not considered in this Appendix, although they should be addressed in a practical implementation of a fitting algorithm.

All the fitting criteria deal with the distance of the measured data points to the substitute feature. If $\boldsymbol{p}_i$ is the ith observed data point, then define:

$$e_i(\boldsymbol{b}) \doteq \pm\min_{q} \{ \, |\boldsymbol{p}_i - \boldsymbol{q}| : f_b(\boldsymbol{q}) = 0 \, \}$$

e_i is the orthogonal distance from the observed point $\boldsymbol{p}_i$ to the surface of the substitute feature. The sign of e_i is chosen to correspond to the sign of $f_b(\boldsymbol{p}_i)$. That is,

$$\begin{aligned} e_i(\boldsymbol{b}) &> 0 \text{ when } f_b(\boldsymbol{p}_i) > 0 \\ e_i(\boldsymbol{b}) &= 0 \text{ when } f_b(\boldsymbol{p}_i) = 0 \\ e_i(\boldsymbol{b}) &< 0 \text{ when } f_b(\boldsymbol{p}_i) < 0 \end{aligned}$$

It should be noted, that if the feature is of perfect form, there exists a value of $\boldsymbol{b}$ for which $e_i(\boldsymbol{b}) = 0$ for all i. In that event, all of the fitting criteria discussed herein result in the same substitute feature. In practice, this situation may appear to exist when the errors in the actual feature are smaller than the resolution of the measuring device.

D1 L_P-norm ESTIMATION

The objective for L_P-norm estimation is to determine the parameters of a substitute feature that minimize the sum of the Pth power of the absolute deviations between the surface of the substitute feature and the observed values. The L_P-norm estimation problem is defined as finding the values of the feature parameters $\boldsymbol{b}$ which minimize

$$L_P(\boldsymbol{b}) = \left(\frac{1}{N}\sum_{i=1}^{N} |e_i(\boldsymbol{b})|^P\right)^{\frac{1}{P}}$$

The "best fit" substitute feature is the one which minimizes the L_P-norm.

D1.1 Least Squares

When $P = 2$, the L_P-norm estimation problem is known as normal least squares or orthogonal distance regression. The term *least squares* is the usual term in the coordinate metrology community.[1] Least squares fitting can be formulated as the optimization problem:

[1] It should be noted that outside the field of coordinate metrology, the term *least squares* usually denotes a different objective from the approach presented herein.

$$\min_{b} \sum_{i=1}^{N} e_i^2 (b)$$

When the values $e_i(b)$ are linear in b, the L_P-norm estimation problem is also known as the total least squares problem.

D1.2 Minimum Zone

When P approaches infinity, L_P-norm estimation becomes minimum zone fitting. Mathematically, as $P \rightarrow \infty$, we have

$$\lim_{P \rightarrow \infty} L_P(b) = \max_{i} | e_i(b) |$$

Finding the minimum zone fit is finding the parameters b which minimizes the maximum magnitude error. This is sometimes called the two-sided minimax fit. (See section D2 for one-sided minimax fits.)

The minimum zone fit is often used in applications that require the substitute feature to be as close as possible to the observed data points. This situation can be formulated as the optimization problem:

$$\min_{b} \max_{i} | e_i(b) |$$

D2 ONE-SIDED MINIMAX

The one-sided minimax approach is often used in applications that require the substitute feature either to contain every observation point p_i or to contain none of the observation points. This situation can be formulated as a constrained optimization problem:

$$\min_{b} \max_{i} | e_i(b) |$$

subject to either

$$e_i(b) \geq 0, \qquad i = 1, \ldots, N$$

when all observed points are to be on the side of the feature corresponding to $f_b \geq 0$, or

$$e_i(b) \leq 0, \qquad i = 1, \ldots, N$$

when all observed points are to be on the side of the feature corresponding to $f_b \leq 0$.

D3 MINIMUM CIRCUMSCRIBING AND MAXIMUM INSCRIBING METHODS

Alternative circumscribing and inscribing methods exist for features of size. Although these alternative methods appear to be very similar to one-sided minimax methods, they are very different. The objective of the circumscribing method is to minimize the size of the substitute feature while keeping all the observed points p_i inside the substitute feature. Similarly, the objective of the inscribing method is to maximize the size of the substitute feature while keeping all the observed points p_i outside the substitute feature.

The substitute features generated by these methods are usually different from those created by the one-sided minimax methods. However, a relationship does exists between these methods. The size of the inscribed minimax feature is not larger than the size of the largest inscribed feature. Similarly, the size of the circumscribed minimax feature is not smaller than the size of the smallest circumscribed feature.

D3.1 Minimum Circumscribed

The minimum circumscribed feature is determined as a substitute feature which has the smallest size $R(b)$ yet contains all the observed data points. This is the constrained optimization problem:

$$\min_{b} R(b)$$

subject to the constraints

$$e_i(b) \leq 0, \qquad i = 1, \ldots, N$$

D3.2 Maximum Inscribed

The maximum inscribed feature is determined as a substitute feature which has the largest size $R(b)$ yet contains none of the observed data points. This is the constrained optimization problem:

$$\max_{b} R(b)$$

subject to the constraints

$$e_i(b) \geq 0, \qquad i = 1, \ldots, N$$

Additional constraints must be added to ensure the range of substitute features considered are reasonable. Without these additional constraints, the maximum inscribed feature is an infinitely large feature with its center or axis infinitely far away from the observed data points.

For example, a circle or a sphere containing observed data points which enclose the desired substitute feature can be stated as requiring the center $\boldsymbol{c}(\boldsymbol{b})$ of the substitute feature to be inside the convex hull of the observed data points:

$$\boldsymbol{c}(\boldsymbol{b}) = \sum_{i=1}^{N} \lambda_i \boldsymbol{p}_i$$

$$\sum_{i=1}^{N} \lambda_i = 1$$

$$\lambda_i \geq 0, \qquad i = 1, \ldots, N$$

D4 OTHER APPROXIMATIONS

Some implementations use the representation function values $f_b(p_i)$ directly instead of the distance function $e_i(\boldsymbol{b})$. When the representation function $f_b(\boldsymbol{p})$ is not equal to the distance, then the optimization will produce a different resultant than the methods described in the previous sections. For example, the representation formula for a sphere may be

$$f_b\,(\boldsymbol{p}_i) = (x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2 - r_c^2$$

whereas the distance formula is

$$e_i(\boldsymbol{b}) = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2} - r_c^2$$

The use of the representation formula in a least-squares approximation results in finding the parameter vector $\boldsymbol{b} = \{x_c, y_c, z_c, r_c\}$ that minimizes $\sum f_b^2(\boldsymbol{p}_i)$. In general, this solution will be different from the least-squares vector $\boldsymbol{b}$ that minimizes $\sum e_i^2(\boldsymbol{b})$.

NONMANDATORY APPENDIX E
DATUM REFERENCE FRAMES (DRF)

The construction of DRFs is a fundamental part of the CMS based measuring processes. The approach is to simulate a full-form engagement of the datum features, but is affected by point density limitations and by shortcomings in the mathematical algorithms which attempt to deal with imperfect geometry. Only a perfect, full form engagement of the datum features can create a "true" DRF. A future objective of this Standard is to provide a mechanism for evaluating the quality of CMS-based DRF computation algorithms.

E1 REPRESENTATION

DRFs are defined by ordered collections of datum features. The datum reference frame is a Cartesian coordinate system, whose components, consisting of the origin, three axes and three base planes, are its associated datums.

E2 CONSTRUCTION

DRFs are typically computed using the vectorial properties of substitute features. Alternatively, DRFs can be constructed using the raw point data to build a theoretical representation of the real part for insertion into a theoretical representation of the associated functional gage.

E3 PRECAUTIONS

The quality of DRFs computed by CMSs depends on the form quality of the datum features, the density and spatial distribution of the raw data, the performance of the DRF construction algorithms, and the quality of the algorithm implementation. Because these factors are present, discrepancies usually exist between CMS-computed DRFs and DRFs corresponding to full engagement of the datum features. Under some circumstances, these discrepances may be significant and therefore of concern.

NONMANDATORY APPENDIX F
FUNCTIONAL GAGE SIMULATION

One method of evaluating geometric requirements is to mathematically model a functional gage and calculate whether or not the simulated "gage" will "fit" the part. This is somewhat analogous to a hard gage fitting a part, with allowable relative motion between the gage and part. Often this approach is the only workable solution when both datums and/or multiple features are toleranced with the maximum material condition (MMC) modifiers.

Adequate Functional Gaging simulation requires careful attention to several issues including part representation, gaging process simulation, interpretation of the output, the role of substitute features, and surface sampling methods. In this process certain precautions must be observed.

F1 METHODS OF PART REPRESENTATION

Typically one of three types of analysis is used to represent the "part" calculated from the measurement samples.

F1.1 Point Method

The coordinate data samples are treated as infinitesimal but real part material. Each sample point is investigated as to whether or not it crosses or "interferes" with the mathematical gage boundaries.

F1.2 Ideal Substitute Geometry Method

Ideal substitute geometry is calculated for the features under investigation. In order that the gaging principles are not violated, this substitute geometry is usually a maximum inscribed or minimum circumscribed circle or cylinder. This substitute geometry may also be a least squares fit shifted by a statistical multiplier or to the point of extreme material. Intersections of gage and "part" surfaces indicate interference; thereby, simulating a "No-Go" condition of a functional gage inspection.

F1.3 Higher Order Fitting Method

A higher order surface is fit to the data for each feature to accommodate the variation of form as well as size. Intersection of surfaces indicate interference, thereby simulating a "No-Go" condition of a functional gage inspection.

F2 Gaging Process

The actual "gaging" process is usually a process whereby successive iterations attempt to lessen the magnitude of the interferences between the gage and the "part" until no interference is realized. These iterations are relative movements between the gage and the part representations. Degrees of freedom of the movements are constrained by the datum reference call outs.

F3 OUTPUT AND INTERPRETATION

F3.1 Go/No-Go

The primary output is an accept/reject disposition.

F3.2 Maximum Interference

If the gage can not fit the part, the maximum interference is usually indicated.

F3.3 Number of Iterations

The number of iterations required to either fit or determine a no fit condition are given.

F3.4 Location of Interferences

Coordinate locations of the interferences are listed.

F3.5 Possible Scenarios of Rework to Allow "FITTING"

More sophisticated simulations may indicate measures of workpiece rework to improve chances of gage fitting.

F4 SUBSTITUTE FEATURES

Substitute features may be used in Functional Gaging if they provide adequate information for the analysis. Considerations include sampling density, form error, workpiece tolerances and type of fit.

F5 POINT SETS

Point sets should be retained for Functional Gaging analysis. The entire set should lie within the workpiece tolerance zone.

F6 PRECAUTIONS

F6.1 Information Extrapolation

Functional Gaging simulation should be used as a tool in conjunction with other data analyses to correctly ascertain workpiece characteristics. Remember its major function is to simulate a Go/No-Go gage.

F6.2 Error Allowance

Makers and designers of functional gages consider the tolerance in manufacturing the gages and its effect on part acceptability. Just as there are errors in building a hard gage, there are errors in the simulated gaging processes that should always be considered whether or not they are incorporated into the gaging calculations.

F6.2.1 Measurement System Uncertainty. This error is due to both systematic and random errors of the measuring process and can have a variety of sources.

F6.2.2 Part Sampling Error. This error is due to the measuring instruments ability to only sample the workpiece surfaces whereas a hard functional gage can contact the full functional surfaces of a part.

NONMANDATORY APPENDIX G
REFERENCES

G1 GENERAL INFORMATION

Beizer, Boris. *Software Testing Techniques.* Van Nostrand Reinhold, New York, 1983.

Crowder, H. P., R. S. Dembo, and J. M. Mulvey. On Reporting Computational Experiments With Mathematical Software. *ACM Transactions on Mathematical Software* 5 (1979) 193–203.

Jackson, R. H. F., P. T. Boggs, S. G. Nash, and S. Powell. Guidelines for Reporting Results of Computational Experiments, Report of the Ad Hoc Committee. *Mathematical Programming* 49 (1991) 413–425.

Myer, Glenford F. *The Art of Software Testing.* John Wiley and Sons, New York, 1979.

G2 INFORMATION RELATED TO CMS SOFTWARE

BS 7172-1989, British Standard Guide to Assessment of Position, Size and Departure From Nominal Forms of Geometric Features. British Standards Institute, London, England.

ISO Guide to the Expression of Uncertainty in Measurement. International Organization for Standardization, Geneva, Switzerland, 1995.

AMERICAN NATIONAL STANDARDS FOR DIMENSIONAL METROLOGY AND CALIBRATION OF INSTRUMENTS

The ASME Publications Catalog shows a complete list of all the Standards published by the Society. For a complimentary catalog, or the latest information about our publications, call 1-800-THE-ASME (1-800-843-2763).

ISBN 0-7918-2651-1
9 780791 826515
L07500