The Integration of Inspection into the CIM Environment

Gerald M. Radack*
Francis L. Merat‡

*Department of Computer Engineering and Science
‡Department of Electrical Engineering and Applied Physics
and Center for Automation and Intelligent Systems Research
Case Western Reserve University
Cleveland, OH 44106

Abstract

We describe a CIM system which combines inspection specification and planning with the design process. The system is feature-based. A part is described in terms of instances of various sorts of generic features: form features (e.g., pockets or ribs) and inspection and tolerance features (e.g., callouts). Rules for inspection are incorporated as part of the generic features. Part definitions and data structures necessary to describe the design are discussed. An overall design consists of a set of feature instances and their relations. Individual features can be automatically checked as they are specified by the designer for overt errors (e.g., incomplete information such as a undefined datum). The overall design is then evaluated for correctness of geometry and inspection specifications. In particular, the designer's tolerance specifications are checked for completeness and internal consistency. Once a design is complete, a module generates process and inspection plans by first concatenating the processing and inspection rules associated with individual features, then optimizing the result with respect to the overall cost of inspection. A specific implementation of the above ideas is described.

Introduction

Although a great deal of attention has been focused on integrating manufacturing planning into the design process, little attention has been given to integrating inspection. There are some exceptions. For example, Valisys Corporation has developed a package for inserting GD&T information into a conventional CAD system, checking the inspection specifications, programming inspection machinery and performing analysis of measured data [11]. In this paper, we describe an architecture for a system which assists the product designer by automatically considering inspectability in addition to manufacturability issues during the design process. We will focus on the integration of inspectability checking and inspection planning with design. This paper is a report on work currently in progress.

Context of the Work

This work is part of a larger effort to develop a "Rapid Design System" (RDS) which is being undertaken by the Manufacturing Research Group at the Air Force Wright Aeronautical Laboratory's Materials Laboratory, located at Wright-Patterson Air Force Base. The goal of the project is to be able to economically and rapidly design and manufacture parts in small quantities (down to a quantity of one). This work is being done in cooperation with an on-base design and manufacturing operation, the 4950th Test Wing.

It is recognized that when parts are manufactured in small quantities, flexible automated manufacturing equipment or hand-operated equipment is the most appropriate. This is because specialized tooling is time consuming and costly. A similar observation holds for inspection. One of the important functions of an inspection planning system is to recommend when to use automated inspection and when to use manual techniques.

Equipment

In the following, we will assume that the automatic inspection device to be used is a coordinate measuring machine (CMM). Such a machine allows for the flexible inspection of a wide range of geometric objects without the need to construct expensive special purpose gages. Thus, a computer-controlled CMM is ideal for the small lot sizes for which the RDS is being designed.

A coordinate measuring machine is basically a gantry-type robot with a sensing probe at the end of the robot arm which can accurately sample individual points on a part's surface (See Figure 6). The sensing probe may be either a contact or non-contact probe. A rigid probe such as a Renishaw PH-2 which can only sense the tops or sides of parts being inspected is probably the most common contact probe. More specialized contact probes such as the Renishaw PH-9 have an extra two degrees of freedom in the probe head allowing the probe to be oriented with respect to the surface under inspection. There are a number of other commercially available probes which use optical techniques such as triangulation or ranging (using variable focus lenses). Optical triangulation probes are typically used for specialized inspection tasks such as gaging sheet metal stampings where a high inspection speed is necessary. Optical ranging probes are less widely used and are typically used when very high measurement accuracies are needed. For the purposes of this work we are ignoring overly specific descriptions of automated inspection procedures such as whether the probe should be a contact or non-contact probe and whether the programming is to be in the DMIS language; we assume that the inspection equipment requires only a list of the nominal locations and the corresponding surface normals of the points to be measured for inspection purposes. For this paper, we are ignoring the problems associated with the generation of a collision-free measurement path; this will be the subject of a future paper. It should be noted that with a manual CMM, the operator is responsible for identifying surfaces and avoiding collisions between the probe and the part.

The exclusive use of a coordinate measuring machine for inspection is not typical of most small machine shops. (A survey of five small local machine shops revealed that only one possessed a manual CMM; none possessed a computer-controlled CMM.) Even those shops with CIM centers do not routinely automate inspection tasks. For example, if only the diameter and depth of a hole in a manufactured part need to be inspected it is very likely that such a part would be inspected by hand using a simple micrometer and a depth gage. On the other hand, the position of a hole is relatively easy to verify with a CMM, but is very difficult to verify by traditional means without constructing special gages. Furthermore, most shops do not have many CMMs and do not want to needlessly schedule parts for

0073-1129/90/0000/0455$01.00 © 1990 IEEE

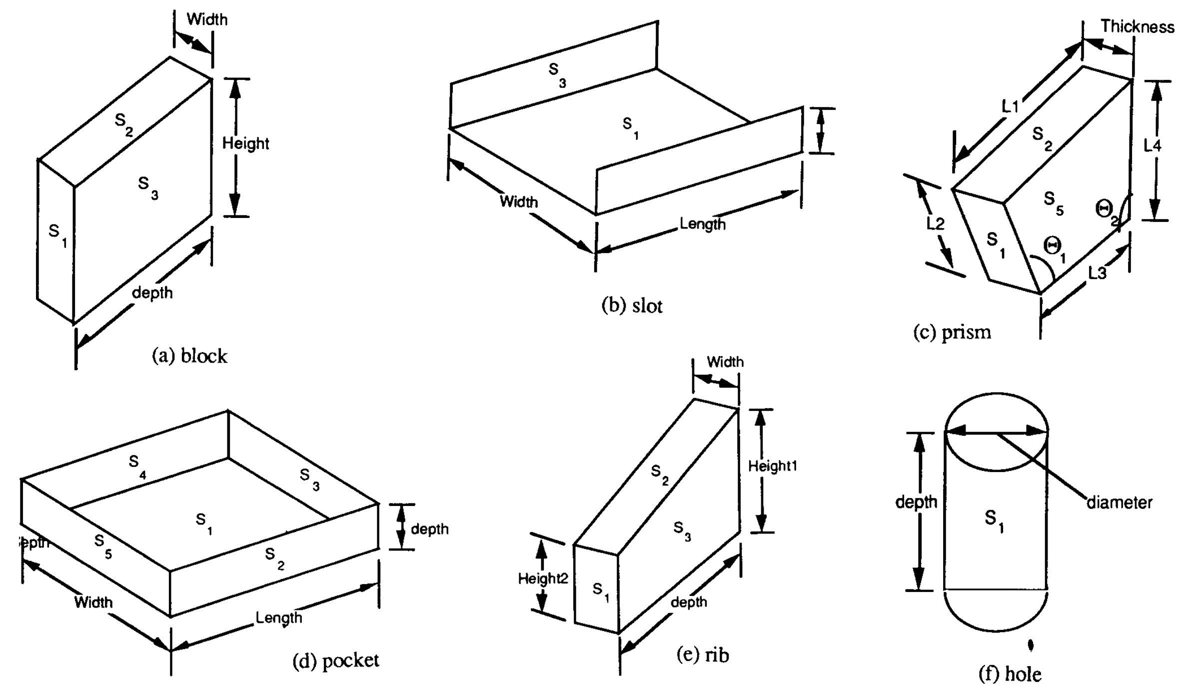


Figure 1 A representative set of form features.

CMM inspection since this could result in unnecessary inspection backlogs. Finally, depending upon the designed part, manual inspection can be faster than automated inspection, as manual procedures do not require jigging or additional measurements to establish a part coordinate frame. The system under development includes information about whether the feature can be inspected using manual or automatic techniques. Manual inspection procedures which are understood by the system, include the use of micrometer plug gages, inside micrometers and micrometer depth gages for hole and slot measurements, micrometers and dial calipers for dimensional measurements, radius gages for fillets, and the use of gage blocks, surface plates and height gages for linear measurements. We are currently interviewing inspectors from the 4950th Test Wing, and are adding to the system additional inspection procedures used by them.

Design Methodology

In this section, we will describe the design methodology that the system is intended to support.

It is generally acknowledged that the current generation of CAD systems is too "low level" and does little more than automate the drafting process. In many cases, creating a computer model is far more tedious than making paper drawings. The payoff, of course, is in the ability to rapidly change a design and to run various analysis tools. Feature-based design reduces the tedium of creating a computer model by allowing the designer to design in terms of form or functional features. A single feature may correspond to many geometric primitives that would have to be entered in a traditional system. Smithers [8] provides further arguments for the need for higher level design tools.

This work was supported in part by the U.S. Air Force under contract F33615-87-C-5250.

Design by Feature

Design by feature is a paradigm which enables a designer to specify an object in terms of elements which are meaningful to him. These elements can specify the function, form, tolerances, manufacture, finish, or inspection of the object. These elements are called "features." In traditional CAD systems, all design information is represented in terms of geometric and topological primitives (points, lines, surfaces, vertices, edges, faces, etc.) and text. A feature-based design system allows the designer to express his true intent, rather than forcing him to map his intent into the aforementioned primitives. This high-level representation of a design in terms of features enables the system to reason about the design. This reasoning can take many forms. We will restrict our attention to reasoning related to inspection issues.

The term "feature" can refer to any component of a design or any attribute of a part. Among the types of features which are present in a design are: form features, geometry features, dimension and tolerance features, and inspection features.

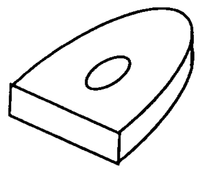
Form Features

Form features define the form or shape of the part. Typical examples of form features include holes, pockets, ribs, bosses, slots, chamfers, and fillets. Many of these features can be further classified into subtypes. For example, holes can be blind or through, countersunk or not, with cone-shaped, flat, or spherical bottom (if blind). Form features can replace the geometric primitives (e.g., lines, curves, surface patches) used in traditional CAD systems to describe the shape of a part.

Figure 1 shows a representative set of form features.

In general, a single form feature will replace several geometric elements. For example, a "blind hole with flat bottom" feature would replace two geometric elements: a cylindrical surface and a flat disk. This does not include any fillets. In traditional systems, fillets and chamfers are normally not represented explicitly as geometric elements. They are added to a drawing as textual notes.

Figure 2 shows how a part might be represented using form features.



(a) A perspective drawing of a part

Feature 1 Foot
- Position
- Orientation
- Width
- Depth
- Radius of curvature

Feature 2 Hole
- Type thru
- Position
- Radius

(b) A possible feature-based part representation

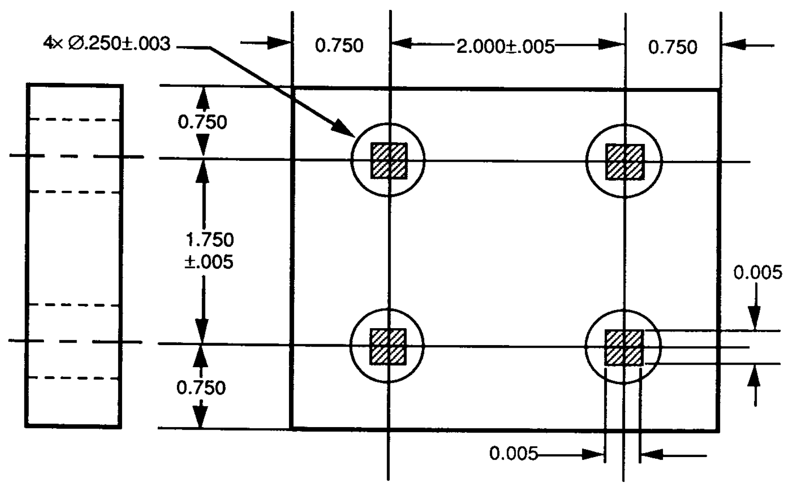
Figure 2 A sample part represented in a feature-based CAD system.

Dimension and Tolerance

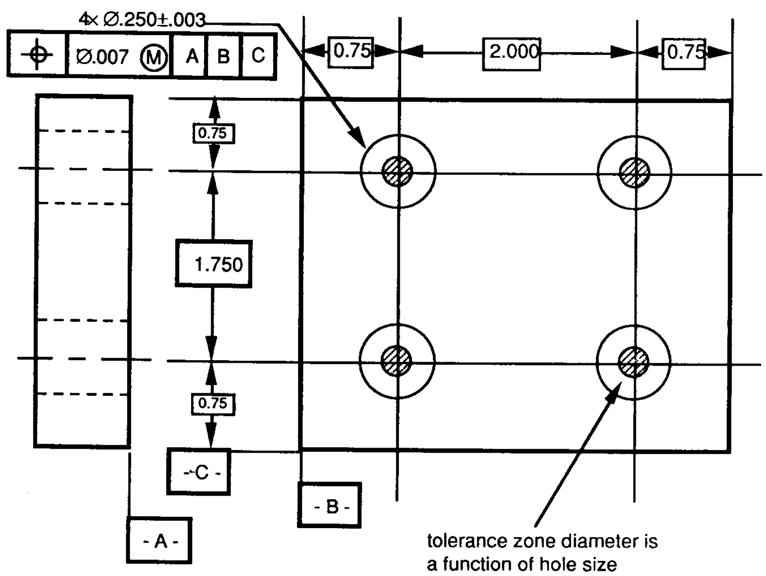
There are various formalisms for dimensioning and tolerancing mechanical designs. Most designers have used the ± dimensioning as shown in Figure 3(a). This ± convention is being supplanted by a standard (ANSI Y14.5, [2]) which is becoming the notation of choice for dimensioning and tolerancing in engineering diagrams. The notation specified by the ANSI Y14.5 standard, usually called "Geometric Dimensioning and Tolerancing" or, more simply, GD&T, is intended to specify unambiguously the allowed variation in shape of a manufactured part; this is not true of ± dimensioning [7]. More fundamentally, GD&T provides a richer framework than conventional ± dimensioning for the designer to indicate functionality by precisely specifying a relationship to be measured. The reader wishing more information about GD&T is referred to the standard [2] or to any of the available textbooks such as [4]. A simple example of ± dimensioning of a block with four holes is shown in Figure 3(a). Note that the dimensions of the hole pattern are given as nominal dimensions with an associated ± tolerance defining the positional tolerance of the hole centers as square regions 0.005 inches on a side. The ± system is ambiguous in terms of the measurement procedure, often leading to inspection confusion. One reason for this is that datums (features with respect to which measured dimensions are checked) are not specified. Referring to Figure 3(a), an inspector might determine the center of one hole (which one is not clear) and measure the positions of all other holes relative to that one or he might measure the holes relative to some other reference. These different procedures could produce different evaluations as to whether the part is in tolerance. Figure 3(b) shows the same part dimensioned according to GD&T conventions. There are several major differences between the tolerance specifications. The linear dimensions between the holes (indicated in the boxes) are exact (or "true") values given without tolerances. The positional tolerance of the holes is explicitly stated in the set of three connected boxes (this is called a "callout" in GD&T) under the hole diameter specification. The first box (the circle with crosshairs) is the GD&T symbol for a positional tolerance. This tolerance is further indicated to be a circular tolerance zone (indicated by the ∅ in the next box) of diameter 0.007 inches. The circled "M" describes how the tolerance specification is evaluated. Finally, the symbols "A," "B," and "C" indicate three mutually perpendicular datums with respect to which the hole's dimensions and tolerance specification are evaluated. In GD&T the position specification of the holes is distinct from the positional tolerance specification of the holes. Aside from more explicitly specifying what is to be inspected (the position of the holes in this case) separating exact dimensions from tolerances is an asset for computer implementation of GD&T and its interaction with modern CAD systems.

GD&T was developed to allow inspections to construct physical gages to make simple go/no-go part inspection decisions. In the case of the block with four holes shown in Figure 3(b), the distances between the holes and the positional tolerances of the holes are both specified. We might envision the inspection gage then consisting of a flat plate with four pegs mounted in it. This gage would then manually be inserted into the part shown in Figure 3(b) and, if the pegs fit into the holes, the part is acceptable; otherwise, the part is rejected. Notice that there are two interacting dimensions in the positional tolerance: the actual position and the diameter. The nature of this interaction is determined by the circled M in the positional tolerance callout. As shown in Figure 3(b), the circled M indicates "maximum material condition, " or MMC and indicates that the tolerance is to be evaluated with the hole diameters being at their minimum diameter, which results in maximum material remaining in the block, hence the name. The minimum hole diameter (and the gage plug size) corresponding to MMC is 0.250 – 0.003 (the diameteral tolerance) = 0.247 inches. This does not include the positional tolerance. In this case, the positional tolerance is a 0.007 inch diameter which means that the actual hole (of size 0.247 inches) can be displaced up to 0.0035 inches in any direction from the specified center location of the hole. The gage plug size must then be reduced by this diameter 0.027 – 0.007 resulting in a gage plug size of .240 inches. Notice that these pins in the gage are the the exact positions as specified in Figure 3(b). It is the callout which contained the specifications for determining the gage pin diameters to be 0.240 inches. Note that this gage was designed for a worse case situation. When the hole size is actually larger than 0.247 inches the specified tolerancing will act so as to increase the acceptable positional variation of the hole. Consider a hole of size 0.250 inches. This hole is 0.003 inches larger than the minimum diameter resulting in an additional 0.003 inches positional tolerance, i.e. 0.007 + 0.003 = 0.010 inch positional tolerance. This is known as a "bonus" tolerance since it can be used to reduce the positional tolerance as indicated above. This "bonus" tolerance does not change the dimensions of the physical gage but must be used to inprepret dimensionial measurements where a physical gage was not employed. An interaction between the positional and diameteral tolerance specifications may not be desired. This can be specified using the RFS modifier (indicated by a circled "S") which stands for "regardless of feature size" and indicates that the positional tolerance is to be evaluated without any dependence upon the hole diameter, i.e. without allocating any bonus tolerance. The concept of bonus tolerances creates the concept of virtual size or virtual condition of a feature. The virtual condition of a feature is the effective size of the feature that must be considered in determining the clearance between mating parts or features and is the most extremene condition of assembly at MMC. The virtual size of the hole in this case is then 0.250-.003-0.007 = 0.240 inches and represents the minimum hole profile to be used in determining assembly clearances.

Present CMM software is excellent for interpreting a selected set of features. For example, a simple command will



(a) A block with holes dimensioned according to the ± convention



(b) A block with holes dimensioned according to GD&T

Figure 3 A block with holes dimensioned according to (a) ± convention; (b) GD&T.

interpret a set of measurements as a hole and will automatically compute the position and diameter of the hole from the measurements. The CMM community has developed a standard software language called DMIS (Dimensional Measurement Interface Specification, [1]) which allows the user to define the nominal points at which an object is to be inspected (i.e. measured by a CMM probe) and then automatically interpret those measurements in terms of a user defined tolerance zone. DMIS does not interpret measurements in a strict GD&T sense, e.g. allocating bonus tolerances in interpreting measurements, since it is only a CMM programming language. It does not allow for the specification of interacting tolerances, which are commonly found in engineering designs (MMC is the most commonly used modifier in designs).

Since our inspection features are based on GD&T, we will describe a subset of GD&T which is presently being implemented in the RDS. All inspection features are known as callouts and correspond to the part specifications shown in boxes in Figure 3(b). Inspection features are subdivided into dimensions, datums, and tolerances. A dimension is an intrinsic property of a feature or a relationship such as a distance between features. A datum is an ideal reference line or plane which is used in the evaluation of a feature measurement. Datums can be located by appropriate datum staging which makes physical contact with the high points of the reference surface (refer to the ANSI standard for a precise definition) or approximated from CMM measurements. Appropriate tolerances for the basic form features

diameter
position
straightness
angle
parallel
perpendicular

Figure 4 Implemented tolerances appropriate for the form features of Figure 1.

shown in Figure 1 are listed in Figure 4. For example, diameter refers to the diameter of a hole feature, position refers to the location of any feature, straightness defines a maximum deviation from the center of an axis (straightness can apply to either an edge of a block, rib or prism feature or to the surface of a cylinder), angle is used to specify the angle between surfaces, perpendicular is a specific instance of angle where the angle is assumed to be 90°, parallel is another specific instance of angle where the angle is assumed to be zero. In general, these implemented callouts either tolerance an intrinsic property of a feature (such as straightness or diameter), or a relationship between features such as a dimension. Relationships between surfaces belonging to an individual feature may be controlled by intrinsic attributes of that feature or by explicit GD&T callouts; relationships between surfaces belonging to different features must be specified using GD&T callouts.

Design Features

Form features and dimension and tolerance features are more convenient to use than the traditional geometry and text elements found in a conventional CAD system. However, designing in terms of such features may still not be the most efficient from the perspective of either the designer or the system. For example, the designer may often specify holes for a particular fastener type. He or she will ususally use the same pattern of GD&T callouts for the same fastener, perhaps only changing a position tolerance or datum reference. Thus, there should be a way to add the form feature for the hole and the GD&T callouts to the model in one operation. However, it must also be possible to modify such callouts once they are added to the model. A macro facility could be used for this purpose, but there are significant advantages to representing such an agglomeration of features as a single "higher-level" feature. From the designer's standpoint, the advantage of this approach is that he or she can move or modify such a higher-level feature as a unit after it has been placed. From the system implementer's point of view, the advantage is that we can incorporate additional intelligence with the higher-level features. For example, we could have an inspection plan covering several dimension and tolerance features simultaneously in less time than the sum of the times of the procedures for the individual features. We will call such higher-level features "design features."

Part Representation

A part is represented primarily as a collection of design features, although the designer does have the freedom to add individual geometry or dimension and tolerance features to the model in circumstances when there are no appropriate design features available. We will assume that a design feature contains a reference to a single form feature and one or more inspection features. Note that a form feature may refer to more then one geometry feature. Each feature type is represented by a record having the following sections: parameters, geometry, and inspection. We will describe each of these sections below.

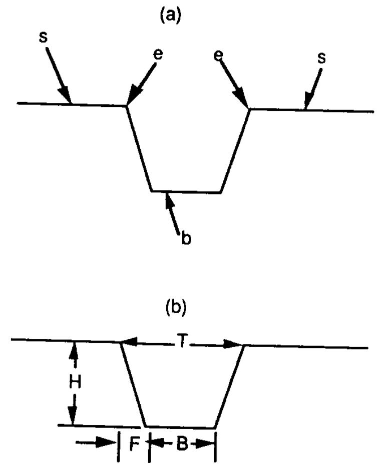


Figure 5 A general through-slot (cross sectional view)

Parameters

A feature can be thought of as a class of shapes. Members of such a class can be characterized in terms of geometric parameters. As an example, consider the case of the through-slot shown in Figure 5(a). Our slot feature is defined so that the bottom of the slot (labelled "b") is parallel to the surrounding surface (labelled "s"). We can then use the following parameters, which are illustrated in Figure 5(b):

- height (H)
- width of top (T)
- width of bottom (B)
- offset from bottom to left side (F)

These are the attributes for this example design feature. It is then necessary for the designer to describe how to inspect the slot by adding the necessary callouts. In addition to the above intrinsic parameters, it is necessary to specify the relationship between a feature and other features in the model. Each feature has parameters that allow it to be positioned with respect to other features in the model. Additional non-geometric parameters may also be specified.

Geometry

The geometry section contains code to generate the geometry features based on the values in the parameter section. A detailed description of this code is beyond the scope of this paper.

Inspection

The inspection section contains information as to how to inspect the feature. In general, there are several ways to inspect a given feature. Therefore, this section contains a list of records, each corresponding to a different inspection plan. Each record contains the following information:

- inspection device,
- preconditions,
- inspection instructions,
- time needed to execute the above instructions, and
- validity conditions.

The preconditions are conditions which must be met before the plan can be executed. For example, in the case of CMM inspection, the side of the part containing the feature must be facing up in order for the probe to be able to reach it. Validity conditions state when a particular plan can be used. For example, in the case of the through-slot, if the bottom width is greater than the top width, a CMM with an ordinary probe cannot be used, since the probe will not be able to reach around the lip of the slot ((e) in Figure 5(a)). The inspection instructions are the actual procedures used to carry out the inspection.

All of the above items except for the inspection device are functions of the values in the parameter section.

Inspectibility Checking

We assume that any part that can be manufactured can also be inspected, given enough time and the proper equipment. Thus, the goal of inspectability checking is to verify that a part can be inspected *economically.* Consequently, inspectability becomes a relative term related to some objective function such as the time required to inspect, or the cost. Using a very simple accounting scheme, we consider the cost of an inspection operation to be the time to execute the operation multiplied by the cost per unit time of the equipment and/or personnel needed to perform the operation.

Since inspectability information is intended as feedback to the designer, it is important that this information be provided rapidly, even at the cost of some accuracy. Ideally, the information should be provided in "real time" during the design process. Because of this, the following heuristics are used for inspectability checking as featùres are added to the model.

1. If none of the validity conditions for the inspection procedures of a given design feature are met, an error is reported.
2. If the preconditions of all inspection procedures for a newly added design feature conflict with those of existing design features, a warning message is given. For example, if there are blind holes on opposite sides of the material, a warning that the material will have to be turned over will be issued.
3. If a newly added feature will require the use of a new inspection device not currently needed, a warning is issued.
4. If, due to the parameters chosen for a design feature, an expensive inspection procedure must be used, a warning is issued.

Some types of inspectability checks are primarily applicable to automated inspection using a CMM, e.g. whether the available probe permits the inspection of a feature such as a horizontally oriented hole. Since the basic machine structure is that of a gantry robot (Figure 6) automated measurements can typically only be taken of the top or sides of an object. In general, a coordinate measuring machine is very applicable to feature-based inspection. For example, a hole can be inspected using the measurement sequence shown in Figure 7. Although five measurement points are shown, a hole is typically measured with only three points, the minimum necessary to fit a circle giving the position of the center of the hole and the diameter of the hole. Larger numbers of data points are used for "least squared error" estimates of the feature parameters. An important aspect of CMM path planning is the stand-off distance—the distance from the surface at which the probe begins a measurement movement. The CMM does not proceed in a direct path from one measurement point to another. Rather, it moves at a relatively high speed from a measurement point to a point at some distance (the stand-off distance) from the surface to be measured. The probe then slowly approaches the surface until the probe contacts the surface. The rationale for this procedure is that a high-speed move quickly positions the probe near the point to be measured, then the probe moves slowly (over a much shorter distance) toward the surface to be measured to avoid probe damage, resulting in a faster overall CMM inspection procedure. Although only a procedure for a hole is shown in Figure 7, similar procedures can be written for blocks, ribs, prisms, slots and pockets.

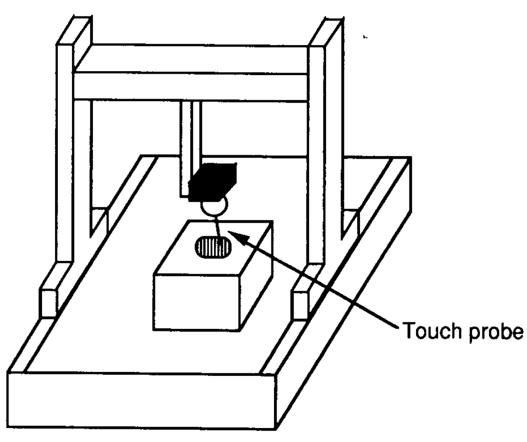


Figure 6 Typical CMM configuration

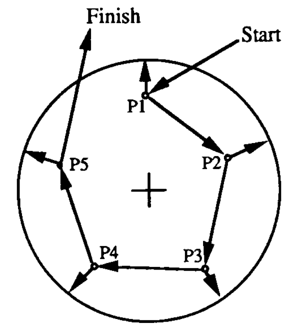


Figure 7 Top view of a single diameter hole with directions for CMM

As an example of an inspection problem, consider a block with different holes to be inspected using a CMM on a number of different faces. In many cases, such a part cannot be inspected without rotating it on the inspection surface. This often requires manual intervention and additional datum specifications, resulting in additional measurements to determine the part coordinate system after each rotation, and increased overall inspection time. The designer needs to be aware of the necessity of defining additional datums in such cases as well as the additional inspection time required for such a design specification.

Another inspection problem has to do with access of the probe to part features for measurement. For example, consider a block with holes located on two adjacent sides and oriented with one hole up. The most common types of measurement probes for coordinate measuring machines are oriented in the vertical direction and can measure top surfaces and non-overhanging side surfaces. Specialized probes may operate in the horizontal plane (but not the vertical). More expensive and sophisticated measurement probes may have an additional two degrees of freedom allowing the measurement probe to be oriented at discrete angles throughout a downward facing hemisphere. Such a probe could be used to measure the sides of a horizontal hole in the sides of the block in the present example. A simple vertical CMM probe can easily measure the hole on the top block surface but could not measure the hole in the side of the block without flipping the block over or changing probe types.

Inspection Planning

Once a part model is complete, an inspection plan can be generated by concatenating the inspection procedures for all features. At present only the most time efficient inspection procedure for each feature is selected for incorporation into the overall inspection plan. This results in an inspection plan which may combine manual and automated inspection procedures in a haphazard fashion. This is acceptable in many aerospace inspection applications where the inspection philosophy is "inspect everything as much as possible." However, an efficient overall inspection plan is possible by considering a more global optimization procedure which minimizes the overall inspection time. This requires information about the setup times for manual and CMM inspection procedures which we have not found in the literature. As a consequence, globally optimal inspection planning has been deferred until actual inspection process data has been collection in a later phase of the project. Inspection plan generation proceeds in two stages: selection of an inspection method for each feature and optimization. In some cases, features must be measured in a certain order for overall inspection plan optimization while in other cases the order does not matter.

A simple cost function for evaluating an inspection plan is then

$$C = \sum_{i=1}^{n} \left[C_{i,\alpha(i)} + (1-\delta(\alpha(i),\alpha(i-1))\, S(D_{i,\alpha(i)}) \right]$$

where:

F_i = feature i;

P_{ij} = *jth* plan for measuring F_i;

D_{ij} = device used by plan P_{ij};

$S(D)$ = the setup time for device D;

C_{ij} = cost of plan P_{ij};

$\alpha(i)$ = the plan number chosen for feature F_i; and

δ is the Kronecker delta function.

In order to minimize the cost of inspection, we need to minimize C with respect to functions α.

The term $(1-\delta(\alpha(i),\alpha(i-1))$ states that we only have to setup a device if it is different from the device used in the previous step of the inspection. Naturally, it would be desirable to reorder the inspection operations so as to minimize the number of setups. Simply grouping all manual procedures and all CMM procedures together can result in a more efficient inspection plan. Other similar groupings which can result in a more optimal (faster) inspection plan include all CMM measurements which are achievable from a single part orientation and (if the inspection evaluation is done sequentially) ordering features such that those which have the highest manufacturing variation (i.e. are most likely to be out of tolerance) are inspected first. This has been described by Galm and Merat [5]. We are presently implementing such a rule-based inspection optimizer as part of the RDS and will describe its performance in a future publication.

The actual location of CMM measurement points is generated for each feature which will be automatically inspected. The nominal inspection points for each feature are described in terms of a feature centered coordinate frame. As the feature orientation and position is part of the feature description, the actual inspection points for each feature are generated in the part coordinate frame by computing the appropriate translation and rotation matrix to transform the feature coordinate frame to the part coordinate frame. Any additional coordinate transformations which further transform the part coordinate frame into the CMM coordinate frame are similarly generated. The resulting overall coordinate transformation is applied to the nominal CMM inspection points to generate the actual inspection points. Note that the CMM inspection plan includes the location of the standoff points and the direction the CMM probe must travel to reach the nominal part surface. This vector must be rotated by the appropriate transformation matrix for the CMM inspection plan to actually work. Consequently, the final CMM inspection program is simply a list of positions and measurement approach vectors (in the part coordinate frame). We have simulated the above procedures in software but are awaiting procurement of our CMM before generating inspection plans for an actual machine.

Inspection Evaluation

Ultimately, the most important result of the inspection procedure is whether the part is acceptable or not. (Other results of inspection might include information that can be used for statistical process control.) The previous discussion of GD&T might have left the reader with the impression that once a design is properly dimensioned and toleranced the inspection plan (including actual inspection points for the CMM) is generated, the inspection points for a CMM or the equivalent manual procedures are automatically generated, the measurements evaluated on a feature-by-feature basis, and the part accepted or rejected on the basis of these measurements. Unfortunately, such a well-defined process only works for parts designed with relatively simple features. There are many subtleties to GD&T when used with CMMs. CMMs only sample a part surface and, as a consequence, cannot be interpreted in the same manner as a hard gage constructed using GD&T definitions. Smooth surfaces can usually be inspected without great concern for where a surface was sampled; rough or complex surfaces demand a great deal of care in the selection and interpretation of the selection points as shown in Figure 8.

Implementation

We have implemented a simple design system incorporating the basic design features described in this paper. Parts of the system have been implemented using SmallTalk V running on an IBM PC clone. Other parts of the system have been implemented in Lisp on an LMI Lisp machine. Both implementations make use of the object-oriented nature of the programming environments. Feature types are represented as object classes. Specific features in the model are instances of these classes.

The specification of callouts and checks for the correct specification of those callouts has been implemented. By "correct" is meant that the callout specification is tested for consistency. For example, a parallelness tolerance should not be applied to surfaces which are perpendicular in the part model. CMM path programs have been generated for a restricted set of designs based upon holes and blocks. We are presently in the process of specifying a more complete set of GD&T callouts as well as integrating all the system components into a complete design and inspection evaluation system which will run on a Lisp machine.

Relation to Previous Work

Turner and Wozny [9,10] presented techniques for tolerance analysis and synthesis. They were primarily concerned with how to allocate tolerances when there are dependencies between tolerances. They used a solid modeling system for manipulation of tolerance information.

Requicha [7] used offsetting techniques to generate tolerance zones based on a notation modeled after a subset of GD&T. An experimental version was implemented as an extension to the PADL-2 solid modeler [3].

Malloch et al [6] describe methods for selecting CMM inspection points so as to reduce the overall number of inspection points without sacrificing inspection reliability. They used offsetting techniques similar to Requicha [7] to generate tolerance zones. They evaluated their measurement point selection using a mean squared error criteria to the original surface as specified by a CAD system.

Conclusions

We have described a feature-based design environment incorporating automatic generation of inspection plans. The manual and CMM inspection procedures for a feature are included as part of the feature definition. The inspection plan for a part can be a mixture of automated and manual inspection procedures. The inspection cost for a part is optimized by choosing a low cost inspection procedure for each instance of a feature and ordering the inspection procedures so as to minimize setup times. CMM inspection procedures for individual features can be concatenated into an overall inspection plan for a part; however, such procedures can be further optimized based upon the selection of inspection points and the ordering of points.

A set of inspection features based upon geometric dimensioning and tolerancing is used to represent tolerancing for inspection planning purposes in the present simple implementation. Inspection features, e.g. callouts, are tested for logical correctness and some limited amount of geometric correctness.

A system incorporating these ideas is currently being implemented.

References

[1] Albert, Mark, "A communications standard for inspection devices," *Modern Machine Shop*, May 1987, 52-63. More detail about DMIS is available from CAM-I Inc., Arlington, Texas.

[2] American Nation Standard ANSI Y14.5M–1982, "Dimensioning and Tolerancing," American Society of Mechanical Engineers, 345 East 47th Street, New York, NY 10017.

[3] Brown, C.M., "PADL-2: a technical summary," *IEEE Computer Graphics and Applications* **2**, 1982, 69-84.

[4] Foster, Lowell W., *Geometrics II*, Revised 1986 Edition, Addison Wesley, Reading, Massachusetts, 1986.

[5] Galm, J.G. and F.L. Merat, "Strategic design driven inspection," *Proceedings of the 1988 International Conference on Computer Integrated Manufacturing*, Rensselaer Polytechnic Institute, Troy, New York, May 23-25, 1988.

[6] Malloch, Charles B., William I. Kwak and Lester A. Gerhardt, "A class of adaptive model- and object-driven non-uniform sampling methods for 3-D inspection," *Machine Vision and Applications* **1**, 1988, 97-114.

[7] Requicha, Aristedes A.G., "Toward a theory of geometric tolerancing," *International Journal of Robotics Research* **2**, 4, 1983, 45-59.

[8] Smithers, T., "AI-based design versus geometry-based design or why design cannot be supported by geometry alone," *Computer Aided Design* **21**, 3, 1989, 141-151.

[9] Turner, Joshua U., and Michael J. Wozny, "Tolerances in computer aided geometric design," *The Visual Computer* **3**, 1989, 214-226.

[10] Turner, Joshua U., and Michael J. Wozny, "A framework for tolerances utilizing solid models," *Third International Conference on Computer-Aided Production Engineering*, Ann Arbor, June 1-3, 1989, 714-723.

[11] Valisys Corporation, 2050 Martin Avenue, Santa Clara, California.

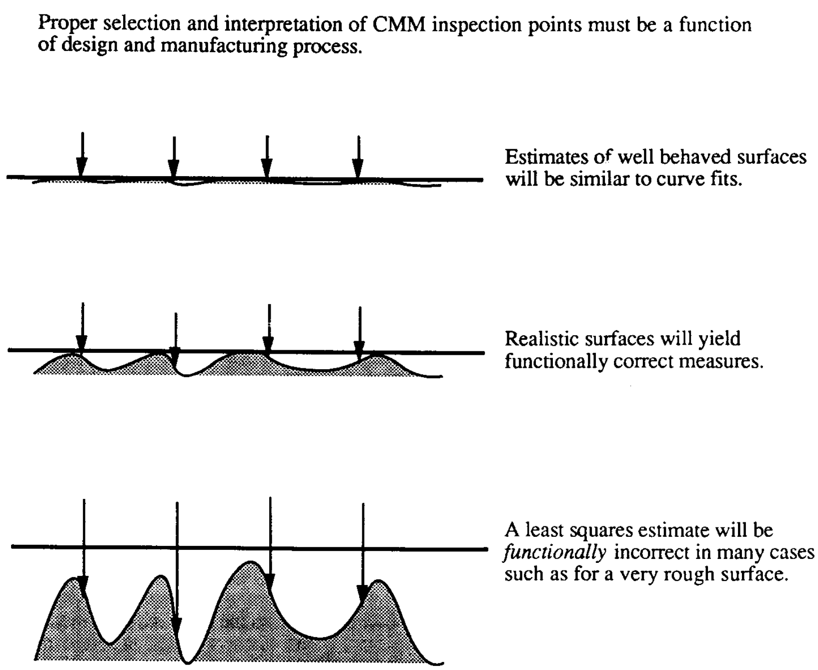


Figure 8 The role of inspection point selection and interpretation in inspection evaluation.