ANALYSIS OF UNCERTAINTIES AND GEOMETRIC TOLERANCES IN ASSEMBLIES OF PARTS

Alan Duncan Fleming

Ph.D. Thesis

University of Edinburgh

1987

Abstract

Computer models of the geometry of the real world have a tendency to assume that the shapes and positions of objects can be described exactly. However, real surfaces are subject to irregularities such as bumps and undulations and so do not have perfect, mathematically definable forms. Engineers recognise this fact and so assign tolerance specifications to their designs.

This thesis develops a representation of geometric tolerance and uncertainty in assemblies of rigid parts. Geometric tolerances are defined by tolerance zones which are regions in which the real surface must lie. Parts in an assembly can slop about and so their positions are uncertain.

Toleranced parts and assemblies of toleranced parts are represented by networks of tolerance zones and datums. Each arc in the network represents a relationship implied by the tolerance specification or by a contact between the parts. It is shown how all geometric constraints can be converted to an algebraic form.

Useful results can be obtained from the network of tolerance zones and datums. For example it is possible to determine whether the parts of an assembly can be guaranteed to fit together. It is also possible to determine the maximum slop that could occur in the assembly assuming that the parts satisfy the tolerance specification.

Two applications of this work are (1) tolerance checking during design and (2) analysis of uncertainty build-up in a robot assembly plan. In the former, a designer could check a proposed tolerance specification to make sure that certain design requirements are satisfied. In the latter, knowledge of manufacturing tolerances of parts being manipulated can be used to determine the constraints on the positions of the parts when they are in contact with other parts.

CONTENTS

SYMBOLS USED

$\subseteq$	Set inclusion
$c()$	A c-function. Also appears as $c_1()$, $c_2()$, ... (page 199)
$C()$	A function of a c-function with the useful property that it is bounded by constants. (page 206)
comp	Set complement
CSG	Constructive Solid Geometry.
E	Extent of a feature or of overlap between two features. (page 61)
E_x, E_y, E_z	Extents of a feature or of overlap between two features measured parallel to coordinate system axes. (page 64)
F	A nominal feature. (page 93)
G	A real feature. (page 93)
H	An extended feature. (page 93)
H' and H''	Congruent copies of an extended feature. (page 94)
l_1, l_2, ...	
m_1, m_2, ...	Offsets used in the definition of tolerance zones where tolerance type is unimportant. $m_1>l_1$, $m_2>l_2$, etc. Usually $m_1>0$, $m_2>0$ etc. and $l_1<0$, $l_2<0$, etc. (page 194)
MMC	Maximum material condition. (page 85)
$\cap$	Set intersection
$O(d;S)$	Offset solid with offset d of solid S. (page 92)
P	Absolute position or MMC-position tolerance parameter. (page 95)
P_1 and P_2	Position tolerance parameters. (page 102)
Q	Orientation tolerance parameter. (page 96)
S_1 and S_2	Size tolerance parameters. (page 94)
sdist	Signed distance. (page 124 and appendix 1)
SUP and INF	Algorithms for evaluating the bounds of an expression constrained by a set of inequalities. (page 65 and appendix 2).

T_f	Form tolerance parameter. (page 94)
U	Set union
x	A vector of DOF_s variables. (page 176)
Δx Δy Δz $\Delta\theta$ $\Delta\phi$ $\Delta\psi$	(Sloppy) degree of freedom (DOF) variables representing the relative position of items when they are free to move. x, y and z indicate translations parallel to x-, y-, and z-axes of a coordinate system and θ, ϕ and ψ represent rotations about these axes. (page 56)
δx δy δz $\delta\theta$ $\delta\phi$ $\delta\psi$	(Rigid) degree of freedom (DOF) variables representing the relative position of items when there is no movement. (page 128)
ΔD δd	General degree of freedom variables. (pages 57 and 156)
ε	Set membership
$\Delta\mu$, $\Delta\lambda_1$, $\Delta\lambda_2$ and $\Delta\lambda_3$	Non-standard translational degree of freedom variables used in the definition of c-functions. $\Delta\lambda_3$ is used for features such as tabs which have distinct surfaces that can make contact. (pages 197 and 198)
$\boldsymbol{\lambda}$	The vector $(\Delta\lambda_1, \Delta\lambda_2, \Delta\theta, \Delta\phi, \Delta\psi)$. (page 199)
∂	The boundary of a set

Chapter 1: **INTRODUCTION**

1.1. GEOMETRIC UNCERTAINTY

Computer models of the geometry of the real world have a tendency to assume that the shapes and positions of objects can be described exactly. However, real surfaces are subject to irregularities such as bumps and undulations and so do not have perfect, mathematically definable forms. The positions of objects are also subject to uncertainty. Engineers recognise this fact and so assign tolerance specifications to their designs to indicate the amount of variation that can be tolerated.

This thesis develops a representation of geometric tolerance and uncertainty in assemblies of rigid parts. Geometric tolerances are defined by tolerance zones which are regions in which the real surface must lie. Parts in an assembly can slop about and so their positions are uncertain.

The geometric variations to be described in this thesis will be referred to as uncertainties. They are not considered to be errors since they are an unavoidable aspect of the real world.

Only small geometric uncertainties will be dealt with. These are typically invisible to a human observer and are small compared with the nominal dimensions of the objects. Two types of uncertainty are considered:

- uncertainty in the shape of parts, usually referred to as "tolerance", and
- uncertainty in the positions of parts.

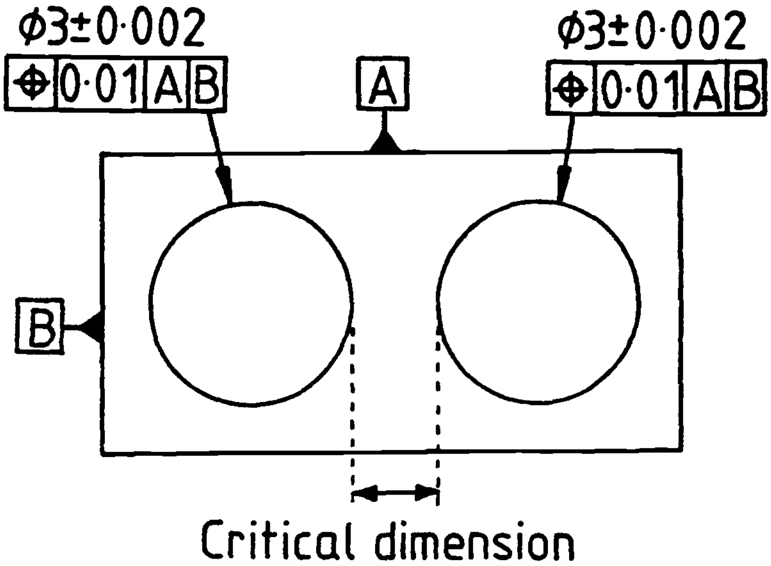
Uncertainty in shape includes variations in dimensions and variations in the form of surfaces. A design specification of a part will often state how much tolerance is acceptable. There are standards used by engineers to define tolerance and a vast amount of knowledge exists about how manufacturing processes affect tolerance. The representations presented in this thesis follow engineering standards as closely as possible.

Uncertainty in position occurs in assemblies of parts where the parts can slop about. The positions of the parts are constrained by the contacts that occur between them but the parts do not fit perfectly. The uncertainty in positions of objects during the planning of assembly by robot will also be considered.

In some situations uncertainties combine to produce larger uncertainties. It is often useful to predict uncertainties which result from the combination of other uncertainties. The following two examples illustrate the sort of problems that can be tackled by the work presented in this thesis.

The first involves the build up of tolerances in a part containing two holes (figure 1.1.1). Using standard tolerancing techniques, to be described later in this thesis (chapter 4) each hole is given a tolerance of position relative to datum faces A and B and each has a tolerance on its diameter. Suppose that, the width of the material between the two holes is critical so that there is an upper and a lower bound on the values which it is allowed. This distance is affected by the uncertainty in position of each hole and by the uncertainty in diameter of each hole. Hence, to verify that the distance will be between the required bounds in all instances of the part all these uncertainties have to be taken into account.

The second example involves checking the possibility of an insertion of a peg, by a robot, through holes in two plates (figure 1.1.2(i)). The lower plate has been placed in a jig by the robot and then the upper plate has been placed on top of the lower

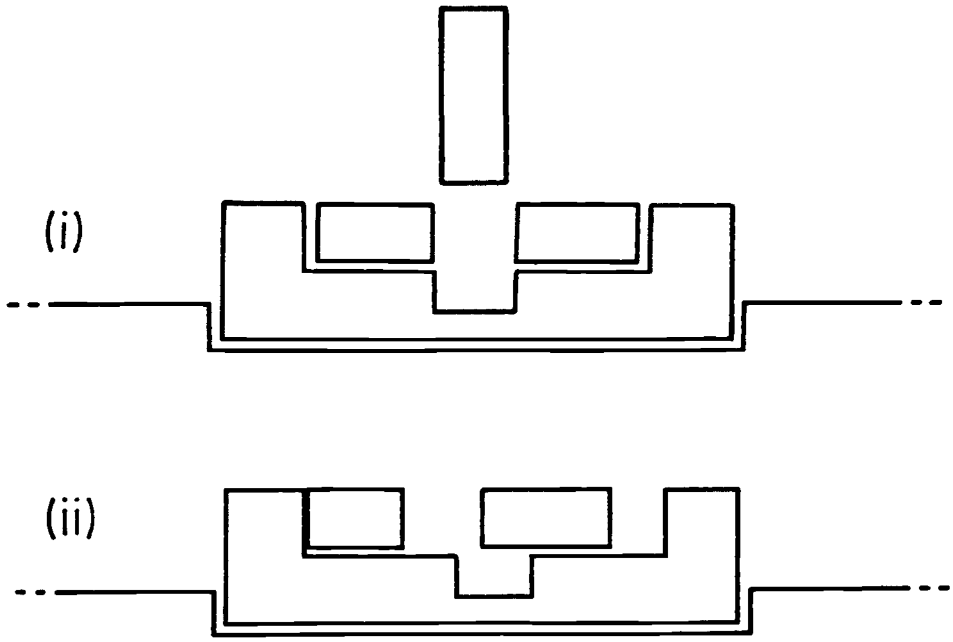


The tolerances in the sizes and positions of the holes all contribute to the uncertainty in the critical dimension. Position tolerance is indicated by the symbol ⌖ and the tolerance on the holes' diameters by the symbol ϕ.

Figure 1.1.1

one. The position of the upper plate is constrained by the edges of the lower plate which is, in turn, constrained by the jig. The two plates form a subassembly and the positions of the parts are constrained by the surfaces which come into contact. The top plate can slop about on the bottom plate by an amount which depends on the exact values of the outer dimension of the top plate and the inner dimension of the bottom plate. If the upper plate has minimum size and the lower plate has maximum size then the slop will be at a maximum. Conversely the slop will be at a minimum if the upper plate has maximum size and the lower plate has minimum size. Since the sizes are unknown (though they may be bounded in a known way) there is a range of possible amounts of slop.

Now suppose that the peg is to be inserted through the holes. The alignment of the holes is important to ensure that the insertion can take place. The uncertainty in the alignment is the



Above, can the peg be guaranteed to enter both holes?
Below, maximum misalignment of the two holes.

Figure 1.1.2

combination of uncertainties in the following:

- The position of the holes in their respective plates;
- The sizes of the plates;
- The position of one plate with respect to the other.

Maximum misalignment of the holes occurs when the following conditions all hold (figure 1.1.2(ii)):

- The top plate has minimum size;
- The internal dimension of the bottom plate is at a maximum;
- The holes are maximally displaced from the centre of the plates but in opposite directions;
- The top plate is at its maximum displacement with respect to the lower plate in the direction which

maximises the misalignment of the holes;

In general, it is not easy to predict the conditions that give rise to greatest uncertainty in a critical dimension or alignment.

In the first of these examples the designer of the part might like to know whether the design was satisfactory. In the second example, it would be useful if a system generating a robot plan could estimate whether uncertainties are going to prevent operation of the plan. It is the goal of this thesis to describe a representation of geometrically toleranced parts with uncertain positions that will allow a system to make such predictions.

Note that there is a reverse problem which can be expressed as the following question. Given the required properties of an assembly of parts, what uncertainty in the shape of the parts can be tolerated? This is an underconstrained problem. It is not tackled in this thesis though a method is suggested in chapter 7 which is a straightforward extension of the work in this thesis.

To clarify the scope of the thesis examples will be given of types of geometric uncertainty which are **not** dealt with. In general these are larger variations and would be visible to a human observer. The following is a list of such types of uncertainty:

- Uncertainty about whether an object or feature of an object is absent or present;

- Variations in position with the same order of magnitude as dimensions of the objects;

- Variations in shape or position which make the presence or absence of some topological condition uncertain (eg. whether or not two surfaces overlap).

- Variations in angle for which trigonometric linearisations (eg. $\sin(\theta)=\theta$) cannot be made.

1.2. TOLERANCES AND THE DESIGN OF TOLERANCED PARTS

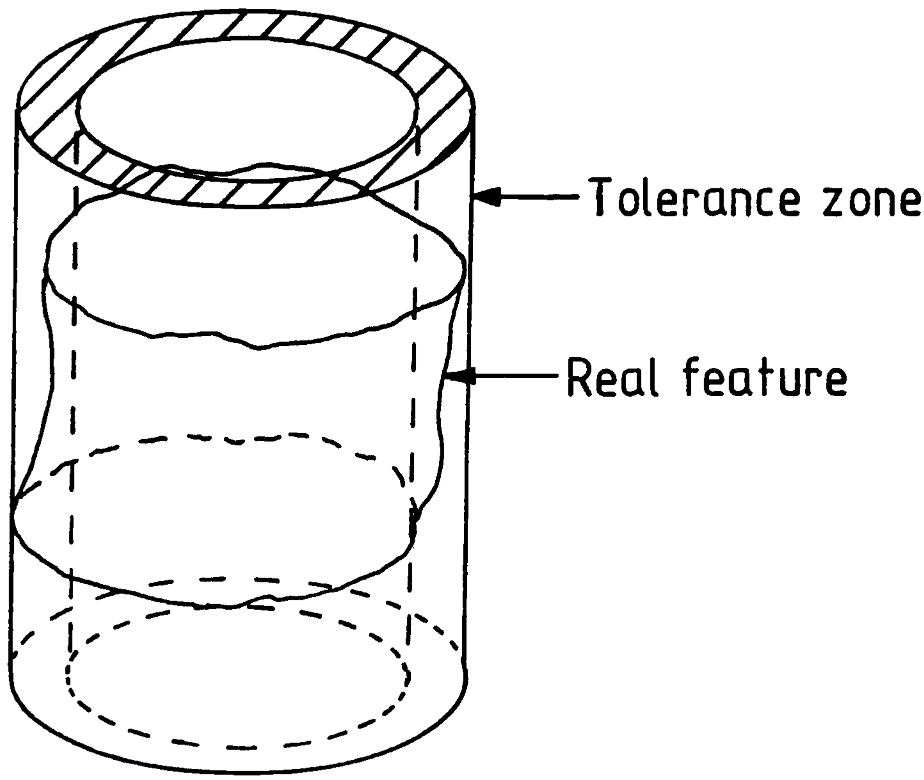
It is impossible to manufacture perfectly formed parts. There will always be inaccurate dimensions and imperfect surfaces. Although the inaccuracies and imperfections might be reduced till they are negligible the production cost would be raised prohibitively. However, the designer can allocate tolerances which state how much variation from the nominal shape can be tolerated if the part is to function satisfactorily.

One method of expressing tolerance is to include an allowed variation beside each dimension. On an engineering drawing such dimensions are indicated by double ended arrows with ranges such as "10±0·01" beside them.

A problem with this type of tolerance is that it is not obvious how to define dimensions on objects which are imperfectly formed and whose surfaces have undulations or roughness. For example, how can the diameter of a hole be defined when it is not exactly circular? To cope with such problems the field of geometric tolerancing was developed. Tolerances are defined by zones in which the actual surface must lie. For example, the form of a hole can be defined, as shown in figure 1.2.1, by saying that its surface has to lie in a zone which is a cylindrical shell with a fixed thickness. The size of the hole could be constrained by fixing the diameter of the cylindrical shell.

The variation in the dimensions of parts can be subjected to a statistical analysis using knowledge of the manufacturing processes. The probability that a part will be outside the design tolerance could be estimated.

Parts can be checked to make sure that they satisfy a tolerance specification with measuring apparatus which is made more accurately than the part itself. Unacceptable parts can be rejected. It is satisfactory if a small number of parts are outside the limits requested at the design stage. This is



A tolerance zone used to define a tolerance on a nominally cylindrical feature. The real surface must be contained in the zone.

Figure 1.2.1

because, it may be more economical to reject a small percentage of parts than to use a more accurate manufacturing process (Michael and Sidall 1981, Parkinson 1984).

A statistical analysis of dimensions at design time would allow estimation of the percentage of parts that will be rejected. However, statistical distributions are often difficult to predict and for this reason they have been not been dealt with in this thesis.

This is equivalent to the assumption that dimensions are evenly distributed over their possible range: they have a square distribution. This is often a reasonable assumption, anyway, for the following reason. A production run will tend to produce dimensions in a distribution which is approximately normal. However due to tool wear different production runs will have different normal distributions associated with them and if the mean varies by an amount greater than the width of the normal distribution then the resultant distribution is approximately square. Bjørke's book (Bjørke), to be discussed in the chapter 2, contains a detailed account of this subject.

The problem of design is to find a tolerance specification that satisfies the constraints of functionality and manufacture. The manufacture of a part involves processes with unavoidable inaccuracies. If the tolerances imposed are too strict then it may be impossible to find processes by which the part can be manufactured or manufactured economically. A machined part is affected by inaccuracies in clamps, jigs and cutting tools. Tool wear means that over time trends occur in the shape of parts produced.

This thesis is mostly concerned, however, with checking that a design satisfies functionality requirements. This is done by assessing the build up of tolerance on each critical dimension. It is not always obvious how individual tolerances contribute to a given dimension. A simple example of this problem was illustrated in figure 1.1.1.

Another requirement of a tolerance specification is that it should be complete and free of redundancy. A tolerance specification would be redundant, for example, if the tolerance on the size or position of some feature of the part were defined in more than one way. A tolerance specification would not be complete if it left the position or size of some feature undefined.

The tolerance on a position or size of a feature may result from the build up of uncertainties from different processes. Therefore, it would be useful to be able to determine the build up of tolerance so as to determine if the resulting part can be guaranteed to satisfy the tolerance specification. A design requirement that two surfaces are to be accurately positioned relative to one another can be most easily guaranteed by holding the part by one of these surfaces while the second is cut.

Suppose that two surfaces are cut during different setups. The uncertainty in their relative position is the combination of uncertainties in the relative positions of the support surfaces used during the setups and the uncertainty in the position of the cutting tool relative to the surfaces of the jig that hold the part.

It follows that in order to check that a tolerance specification is satisfactory it must be possible to assess the combined effects of uncertainty from various sources. This thesis shows how this can be done.

Much of the work described in this thesis deals with assemblies of parts. Some points concerning assemblies of parts are given here to throw light on the problems involved.

In an assembly the positions of parts are constrained by the contacts between them. The exact positions of the parts cannot, in general, be determined because the parts fit together loosely. Instead a set of positions can be associated with each part. This set defines the **position uncertainty** of the part.

To understand an assembly of parts both variation in shape and variation in position need to be taken into account. Variation in shape occurs between different manufactured instances of the same part. This happens for each part in the assembly independently. However, the shape of each part is **fixed** in each **instance** of the assembly (assuming that they are made of a rigid material). On the other hand, the positions of parts are not fixed in a given instance of the assembly.

The set of possible positions of a part in an assembly is affected by the shapes of the individual parts. Hence, a **set of sets** of possible positions can be associated with each part.

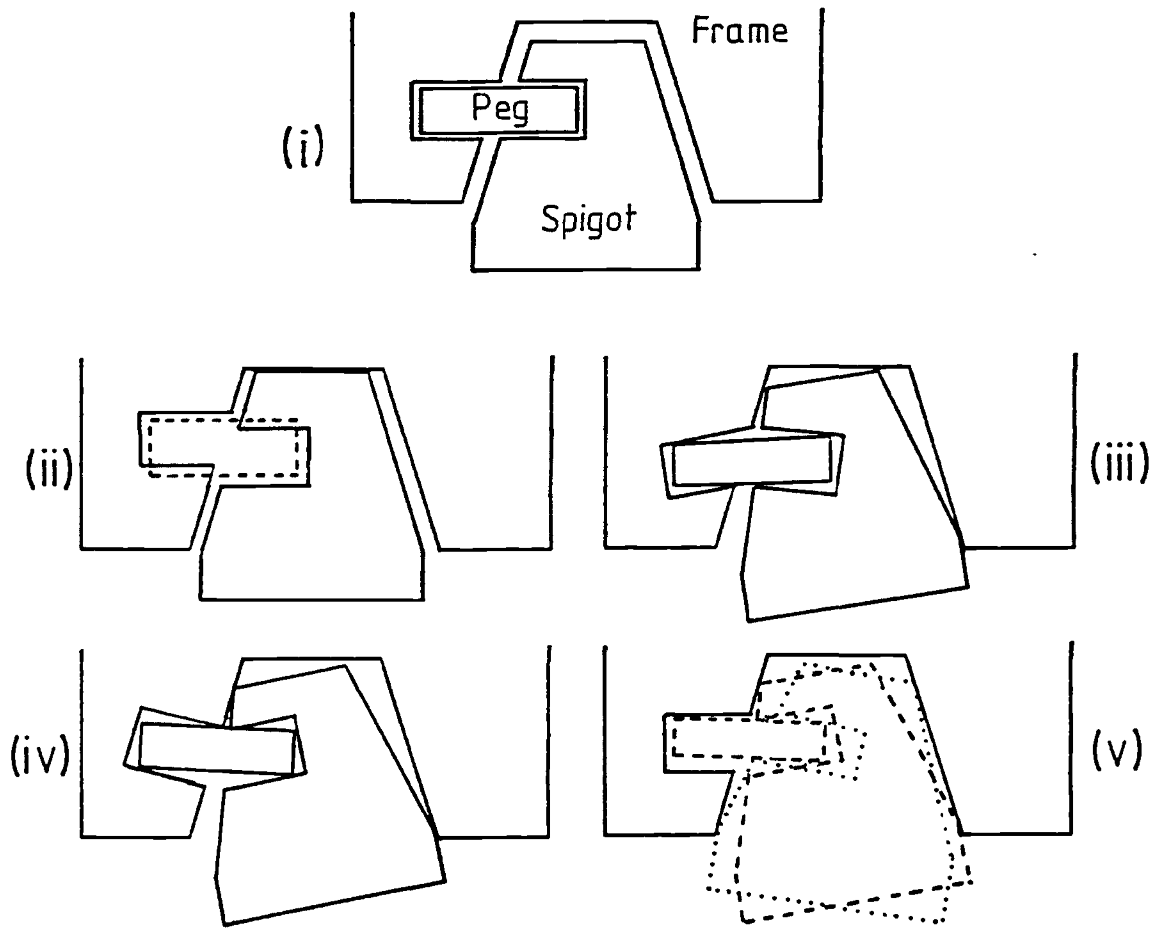
There are several interesting questions that can be asked about a structure of toleranced parts. For example consider figure 1.2.2(i) showing a conical spigot fitted into a frame and held in place by a peg. Since none of the parts fit perfectly into any of the others the spigot will be able to move about inside the frame.

Figure 1.2.2 also shows some problems that might occur if the parts have too much tolerance and these are listed below. These situations might only occur when certain dimensions of the parts are near extremes of their ranges in ways that combine unfavourably.

- Figure (ii). The parts might not fit together.
- Figure (iii). The spigot might be forced to an unacceptable position. In the figure the spigot is forced to be inclined as a result of the holes being inclined in opposite directions.
- Figure (iv). The spigot is able to attain an inclination which might be unacceptable (though it is not forced into this position).
- Figure (v). The maximum possible slop of the spigot in the frame might be unacceptably large.

Chapter 6 shows how such problems can be predicted from a computational representation of the assembly.

Each time an assembly is constructed parts must be selected from the family of non-identical instances of each part. Parts could be chosen deliberately so that they fit together as well as possible or they could be picked at random. Although deliberate choice of parts has the advantage that it allows larger tolerances



(i) The nominal spigot, frame and peg.

(ii) to (v) Problems that can occur when there is too much tolerance on the parts in an assembly.

Figure 1.2.2

this technique is not normally used because of the problem of finding replacement parts. It would be difficult to formalise such techniques.

Assembling parts picked at random is the technique most often used and is the one that will be dealt with in this thesis. As a result dimensions of different parts in an assembly are entirely independent.

1.3. UNCERTAINTIES IN ROBOTIC APPLICATIONS

A robot performing a task is affected by uncertainties both in its own mechanism and in the world around it. When planning a task for a robot it is useful to analyse these uncertainties. The extent to which uncertainties need to be considered depends on the precision of the task. This section shows how the analysis of a static structure of parts is useful in robotics and discusses other ways that uncertainty needs to be dealt with.

Two applications to which robots have been commercially applied are paint-spraying and welding. In paint-spraying small geometric ("non-topological") uncertainty is not a problem due to the low precision of the task. Some welding tasks can be performed satisfactorily by using setups that reduce uncertainty to an acceptable level. In other tasks uncertainty can be overcome by using a sensor in closed loop control. For example, vision sensing has been used satisfactorily to enable a robot to follow the edge of two sheets of metal being welded (Clocksin et al 1982).

In assembly tasks precision becomes more important and therefore the work in this thesis is more relevant to this application. Parts to be assembled are often available from feeders. The manipulator moves the parts from the feeders and assembles them on the workbench. Parts may be held to the workbench in jigs or clamps. Some parts will be placed on top of or inserted into other parts.

There are three causes for uncertainty in the actual positions of parts to be manipulated during robotic assembly. Firstly, the initial positions of parts have uncertainty. Parts are provided by a feeder which will often put constraints on the position of the part. The amount of variation in position of delivered parts depends on the feeder being used and possibly on the set-up of a particular feeder. However, it will often be possible to estimate bounds on the positions in which parts are provided.

Secondly, the position of the robot's end effector is subject to uncertainty. The robot will not go exactly to the commanded position. This type of uncertainty may be known from experience with the robot or from manufacturer's specifications.

Thirdly, the sizes and shapes of parts are variable. Tolerances specify how much the parts can deviate from their ideal shape. Tolerances are usually known from specifications given by the manufacturer of the parts.

All uncertainties in actual positions of parts result as a build up of combinations of these three types. For example, suppose a part is picked up by a robot and then put down somewhere else. The position of the part in the gripper depends on the position of the gripper when the part was picked up and on the position of the part before it is picked up. The position of the part after it is released depends on the position of the part in the gripper and the position of the gripper at the time of release. Hence the uncertainty in the final position of the part depends on the uncertainty in the initial position of the part and the uncertainties in the initial and final positions of the robot. The part will be placed in contact with the workbench, a jig or other parts. An understanding of these interactions is necessary to determine the final constraints on the parts.

The position of a given feature of the part will be affected, in addition, by the tolerance between the feature and the surfaces used to grasp the part. This may not be a dimension which was specified directly by the manufacturer of the part and so must be inferred from dimensions and tolerances that were specified. The work in this thesis can be applied to this problem.

Another major application of this thesis involves the partially assembled structure on the workbench which appears during assembly. When planning an insertion of a part into the structure it is useful to know the uncertainties in the structure so as to determine whether the part can be guaranteed to fit satisfactorily. This requires analysis of the uncertainties in a static assembly.

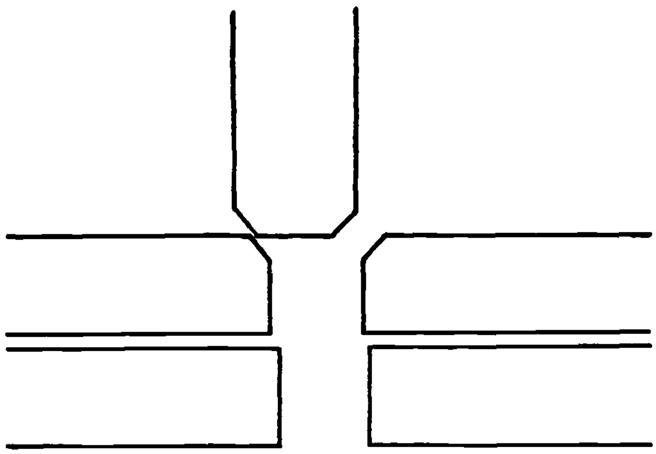
Consider again the example shown in figure 1.1.2 and suppose that the robot approaches holding a peg which is to be inserted through both holes. The robot stops with the peg poised above the hole. We are interested in the uncertainty in the relative position of the peg and the hole. This is the combination of the uncertainty in the following: the position of the gripper, the position of the part in the gripper and the position of the hole relative to the workbench. The latter can be calculated by analysing the dimensions and tolerances of the two plates and the jig.

Now, suppose that the tip of the peg has penetrated the opening of the hole as shown in figure 1.3.1. (Although the peg and the hole are chamfered we are dealing with the moment before the chamfer has become effective.) To allow penetration to occur the upper plate and the peg must satisfy the constraint that the entrance of the hole contains the tip of the peg. The initial constraints must guarantee that this will occur.

The situations before and after penetration can be treated as static structures of parts and thus can be handled by the work in this thesis. The set of possible positions of the plate relative to the peg before penetration must be contained in the set of possible positions after penetration. Therefore, the first situation must be at least as tightly constrained as the second.

In general this thesis deals with the prediction of uncertainty bounds. If the predicted uncertainty bound is too large to be acceptable then uncertainty reducing steps can be introduced into the plan.

One method of reducing uncertainty is by the use of sensors. After sensing the uncertainty in the part's position will be reduced. However there will be uncertainty in the measurement



The tip of the peg has penetrated the entrance of the hole.

Figure 1.3.1

made by the sensor. Hence a sensor introduces uncertainty and to be useful it must be a smaller uncertainty than what was present before use of the sensor. Analysis of uncertainties introduced by a sensor is necessary to determine whether the sensor will provide sufficient uncertainty reduction.

Another method of reducing uncertainty is by making movements so that geometric relationships are created between objects. For example, a part held in a robot's gripper could be moved until it makes contact with some other part which is fixed to the workbench. The relative position of the contacting surfaces is then known with hopefully less uncertainty than previously and from this the position of the gripped object relative to the robot's gripper could be deduced.

An assumption made in this thesis is that the size of uncertainties is predictable. However, in a complex and changing workstation there may be too many sources of uncertainty to keep track of them all. As a result, there may be situations where the size of uncertainties are unpredictable. An alternative method for dealing with uncertainties has been suggested by Dufay and Latombe (1984). Basically this involves learning what variations occur in a robot plan by executing it several times. Initially the uncertainties are assumed large and many sensor readings are made during plan execution. If, after several runs of the plan, a sensor reading is found to be unnecessary it can be dropped from the plan or a less rigorous test can be substituted for it. A difficulty arises if an uncertainty is initially assumed to be smaller than it really is. An initial analysis of the uncertainties would be useful for this reason.

1.4. THE GENERAL APPROACH

In this thesis there are three domains over which uncertainties are analysed:

- Assemblies of perfectly formed parts;
- Single toleranced parts;
- Assemblies of toleranced parts.

The first of these is the subject of chapter 3. Here, there is uncertainty of position but no uncertainty of shape. The representation of position uncertainty has to be investigated. A network is constructed with nodes representing parts and arcs representing possible contacts between pairs of features. From the geometry of each contact a set of inequality constraints is derived and are attached to the associated arc of the network. Techniques are described for analysing such a network of constraints. The result is a set of constraints on the positions of the parts which represent the effect of all possible contacts between them. The basic technique of analysing a network of inequality constraints is used in all three of the domains.

Chapter 4 describes a formalism of geometric tolerancing developed by Requicha (1983a) and makes some comments on this. This is used as a basis on which to build the work of the following two chapters.

In chapter 5 single toleranced parts are dealt with. There is uncertainty of shape but no uncertainty of position. A network is constructed with nodes to represent tolerance zones and datums and arcs to represent relationships between them. The ways that relationships occur are categorised and each category is dealt with in detail. It is explained how constraints can be associated with each arc. The network is analysed to determine constraints on the relative position of chosen features.

Chapter 6 deals with the domain of assemblies of toleranced parts in which there is both uncertainty in shape and uncertainty in position. It is shown how the constraints arising from contacts between imperfect surfaces can be represented. Each part gives rise to its own network of features and datums and the networks are linked by relationships which represent possible contacts. The result of analysing this network is to determine extremal positions attainable by parts and bounds on the amount of slop in the assembly. It is also possible to determine whether the parts can be guaranteed to fit together.

Chapter 2: **RELATED WORK**

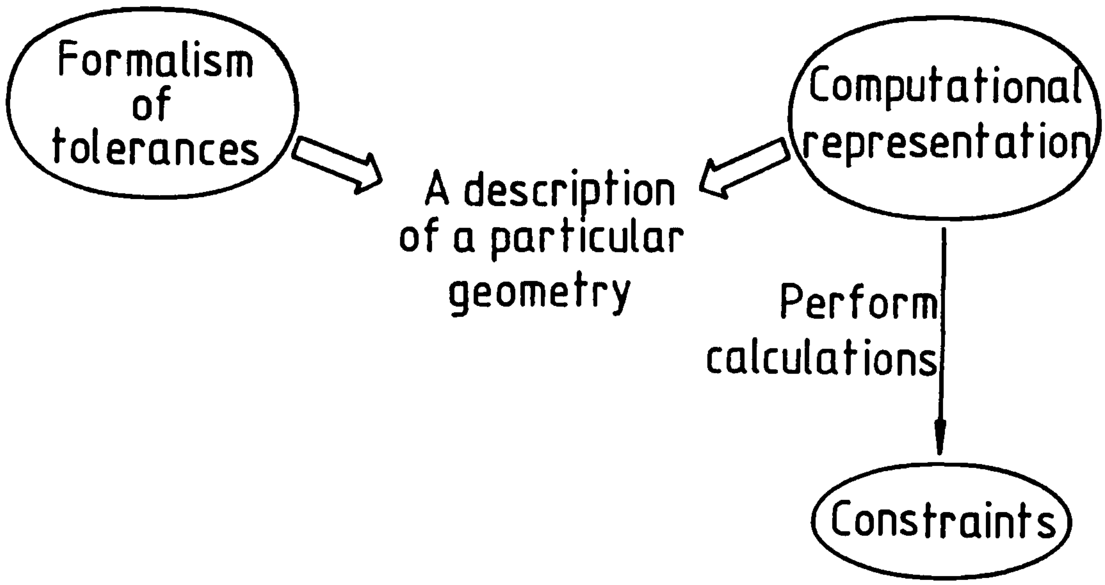
There are two main categories of work described in this chapter. Firstly there is work involving tolerances mostly in computer aided design applications. Most of this takes an allocation of dimensional tolerances (as opposed to geometric or form tolerances) and analyses it to make sure that the design requirements are satisfied. Work by Requicha, however, produces a formalism for geometric tolerances. Secondly, there is work involving uncertainties in off-line robot programming. This involves uncertainty in the positions of objects and in sensors but only deals with uncertainty in shape to a limited extent.

2.1. RELATED WORK CONCERNING TOLERANCES

The work to be described in this section covers various aspects of the tolerance analysis problem. To understand the contribution provided by each piece of work the components that would be involved in a complete computerised system for tolerance analysis are described below and illustrated in figure 2.1.1.

Firstly, the representation must have a clear semantics and be based on a sound formalism. Unless this is done the precise meaning of the representation is not clear. Traditional tolerancing standards are unclear in many respects and contain ambiguity. There are many points to be clarified if tolerances are to be represented computationally. The biggest problems occur in geometric tolerancing but formalisms are also necessary for plus/minus tolerancing and the statistics of tolerances.

Secondly, a system must have a means of representing geometry and its variability. This representation must describe the nominal shapes of objects as well as the tolerances applied to the



A formalism of geometry along with a computational representation describes a particular geometric situation. Calculations performed on information in the representation yield constraints on dimensions or positions of particular interest.

Figure 2.1.1

objects. The computational representation can be interpreted using the formalism. The result is a description of the geometry of a particular geometry along with its variability.

Thirdly, it must be possible to obtain results from the representation. This can be done by performing calculations on the information stored in the computational representation. The validity of these calculations can be checked if the representation is based on a formalism. The results are usually in the form of constraints on dimensions or constraints on positions of features and parts. A method of representing these constraints must be available.

Ideally, analysis of tolerances will take place as part of a larger system used, for example, as a design tool or a robot planning tool. Therefore, it is useful to link a tolerancing system to a general geometric modeller. Up to the present, commercially available geometric modellers include no representation of tolerances though some research is being done on this area. Although some of the work to be described below includes geometric models these have been developed specifically for the tolerancing system involved and are isolated from wider applications.

Bjφrke's book "Computer-Aided Tolerancing" (Bjφrke) discusses many aspects of tolerancing. He produces a useful classification of dimensions and investigates their properties. Some of these classifications will be given in detail below.

Bjφrke makes use of chains of toleranced dimensions which contribute to a dimension of particular importance called a "sum-dimension" because it is usually the vector-sum of the contributory dimensions. Dimensions must form a linear sequence: he does not investigate what happens when parallel dimensions provide contributions to the same sum-dimension. A dimension may be between two features of the same part or may be between features of different parts. Dimensions between different parts may be subject to change as the parts slop about relative to one another. He defines four different binary categorisations of dimensions:

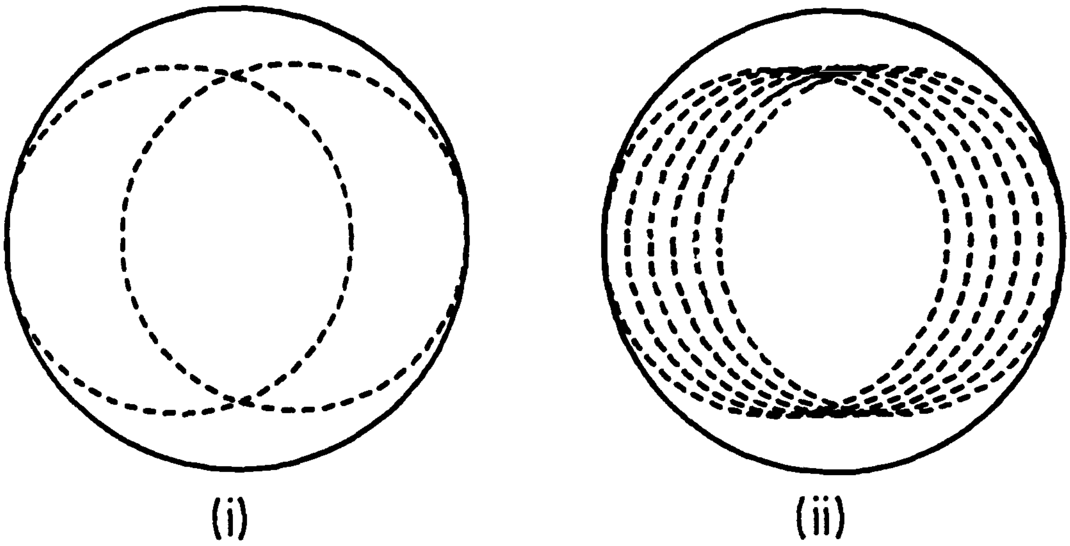
1. Spans and gaps; A span is a dimension between two features of the same part. It has a fixed value. A gap is a dimension between features of different parts. Its value is variable.

2. Line vectors and plane vectors; A line vector is a dimension parallel to the sum dimension. A plane vector is not parallel to the sum dimension.

3. Lumped direction and distributed direction; A plane vector dimension may have a unique direction in which case it is said to have lumped direction. A plane vector with variable direction is said to have

distributed direction.

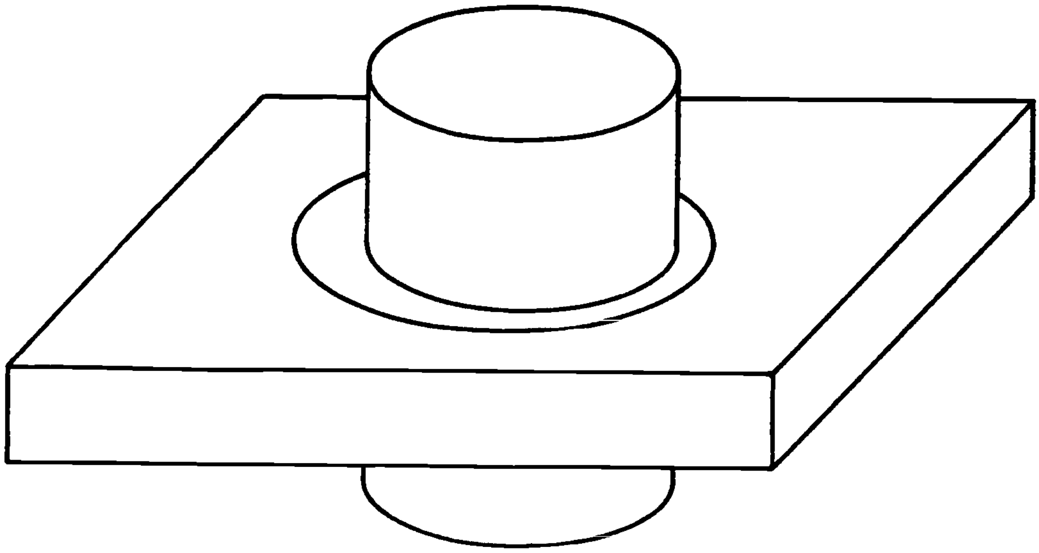
4. Lumped magnitude and distributed magnitude (figure 2.1.2); This is applicable to gaps only. In the first case the dimension between the axes of the peg and the hole has lumped magnitude because some force keeps the peg in contact with one side of the hole. In the second case the peg is completely free to move about inside the hole and so the dimension has distributed magnitude.



(i) A dimension with lumped magnitude.
(ii) A dimension with distributed magnitude.

Figure 2.1.2

Most spans have lumped direction but figure 2.1.3 shows an example of a span with distributed direction. The rod is nominally concentric with the bush but is actually slightly eccentric due to manufacturing error. The bush is welded into a hole in the plate. The dimension between the axis of the hole and the axis of the rod has distributed direction.



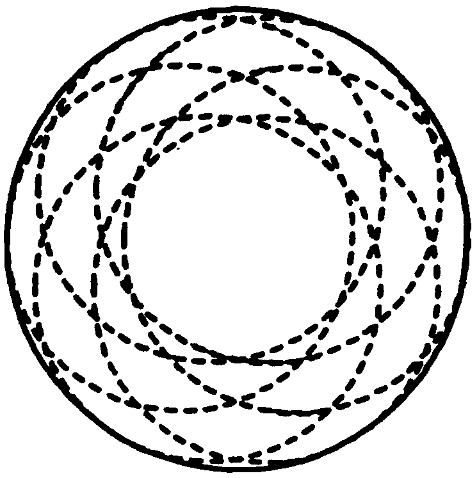
The dimension between the centre of the hole (filled by the bush) and the axis of the rod is a span with distributed direction.

Figure 2.1.3

Figure 2.1.4 shows an example of a gap with distributed direction. The dimension is between the axis of the peg and the axis of the hole.

A chain of dimensions is assessed in one direction only. The contribution of each dimension to the total variation in the sum-dimension is affected by the inclination of the dimension with respect to the sum-dimension. In the case of dimensions with variable direction or magnitude (usually caused by slop between the parts) the relevant component of their total variation must be determined. All variations in the appropriate direction are simply added together.

Bjørke investigates different statistical distributions of tolerances that are likely to occur from manufacturing procedures. Each contributes to a sum-dimension in a way depending on the type of dimension. The types of statistical distribution are classified so that they can be dealt with by a computerised system on a case by case basis. One example of a statistical



A dimension between the axis of a peg and the axis of a hole which is a gap with distributed direction.

Figure 2.1.4

distribution that he investigates is the moving normal distribution. This allows tool wear to be accounted for. Each production run produces dimensions which lie in a normal distribution but the median value and the spread of the distribution varies monotonically with tool wear.

Given a chain of dimensions with specified properties his work allows the build up of tolerances in the chain to be determined. Hence design requirements can be verified from design specifications. Since dimensions and their tolerances have been classified automation of this process becomes feasible.

Bjφrke's analysis of tolerance build up is limited in two ways. Firstly, variation is assessed along a single direction. Variations in angle and interaction between variations in different directions cannot be assessed. Secondly, no differentiation is made between variations in positions of parts and variations in positions of features within a part. This means that the method for determining results about assemblies, such as whether or not parts can be guaranteed to fit together, has not been formalised.

The input to a computerised tolerancing system using Bjørke's formalisations would take the form of specifications of chains of dimensions and attributes attached to each dimension. No mention is given of how such information could be attached to or derived from a geometric model.

Hoffman (1982) describes a representation for the constraints obtained from toleranced parts. An instance of a part is represented as a vector, $\mathbf{x}$, of parameters describing the positions of edges, vertices and faces of the part. There is no geometric model, as such: it is assumed that the model has been reduced to a set of suitable parameters.

A tolerance specification is represented as a set of inequalities involving $\mathbf{x}$ of the form,

$$L \leq f(\mathbf{x}) \leq U,$$

where f is a scalar function and L and U are numbers. However, it is not stated how these inequalities are to be derived from a designer's tolerance specification.

A production plan is thought of as a sequence of processes each of which produces one or more features. Each process is modelled as the set of constraints which the parameters representing the features must satisfy after the process has taken place. The constraints arise from the known inaccuracy of the processes and the inaccuracy of the setup during the process.

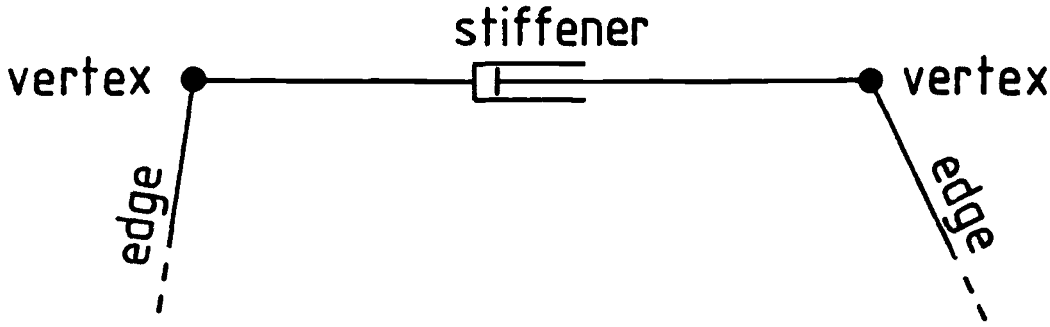
Hence there are two sets of constraints, one representing the intended class of possible parts and the other representing the actual class of parts that would be manufactured. Linear programming is used to determine whether or not the class of parts produced is contained in the class of intended parts.

Hoffman has represented the constraints involved in analysis of a toleranced part in a concise form. However, his assumption that a part can be parameterised means that imperfect form cannot be treated in the same way. He does not mention how the constraints could be derived from geometric models or from a designer's specification. Nor does he say how to convert a production plan into constraints.

The work of Hillyard (1978), in contrast to the work of Bjørke and Hoffman, makes use of geometric models which allow tolerance specifications to be attached. His representation is reminiscent of "boundary representations" commonly used in solid modellers in that the topology of a part is defined by a network of vertices, edges and faces. However, the sizes and positions of the entities are not represented directly. Instead, dimensions are added to the topology in the form of "stiffeners". A stiffener can be attached to vertices, edges or faces according to the type of dimension it represents. They can be imagined as rigid mechanisms attached to the floppy structure of the topology.

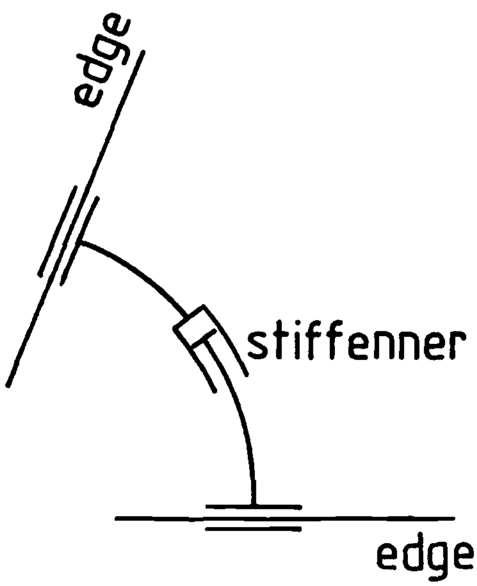
However, there are many dimensions not represented explicitly in an engineering drawing and stiffeners must be present to represent these. For example, the faces which appear to be perpendicular on a drawing should be assumed to be perpendicular unless there is information to the contrary.

An example of a stiffener is a dimension between two vertices represented as a rigid rod whose ends are attached to the vertices. A tolerance can be added to the dimension by inserting a conceptual "piston" into the rod so that its length can vary by a certain amount but so that it remains straight (figure 2.1.5). A toleranced angle can be represented by a curved rod whose ends are attached to two edges and which contains a curved piston as shown in figure 2.1.6. More complex stiffeners are needed in three dimensions. Each stiffener has one or more parameters associated with it. These represent the extension and contraction that the stiffener can make and are bounded above and below.



A stiffener fixing, within a certain tolerance, the dimension between two vertices.

Figure 2.1.5



A stiffener fixing, within a certain tolerance, the angle between two edges.

Figure 2.1.6

Hillyard has produced a classification of all stiffeners required to represent different types of dimensions. He expresses the constraints implied by stiffeners as equations between vertex parameters and stiffener parameters. Thus there are two levels at which the shape of a part is parameterised: by parameters on stiffeners and by parameters on vertices. The constraints on the former follow directly from the tolerance specification whilst constraints on the latter are more useful from the point of view of checking a tolerance scheme though they are not known directly. The two sets of parameters are linked by the equations implied by each stiffener.

Obtaining results from the representation requires inverting the set of equations obtained from all of the stiffeners. This is difficult when there are many parameters involved and when the equations are complex.

In conclusion, Hillyard has described a modelling system that integrates dimensions and tolerances. The method seems to be compatible with widely available boundary representation modelling systems though no discussion is given on such a link. He has classified all types of dimensions that can occur and listed the constraints that each imposes. However, the representation does not enable the introduction of geometric tolerances: all edges and faces are assumed to have perfect form. The constraints obtained make it possible to deduce whether a tolerance scheme is satisfactory.

A problem in his geometric models is that positions of faces are not represented directly. If a face has more than three vertices then there is no plane that contains them all. It is not obvious how the variation in position of a face can be inferred in a consistent way from the variations in positions of its vertices.

None of the work discussed so far has dealt with geometric tolerances. Requicha (1983a and 1983b), however, has developed a formalism of geometric tolerancing that clarifies and generalises standard tolerancing practice. It has allowed the computational representation of tolerances described in this thesis to be developed. His intention was to follow as closely as possible standard tolerancing practice. Ambiguities have been removed and tolerances have been generalised to be independent of feature shape. For example tolerances of flatness and cylindricity have been replaced by a single **form** tolerance which can be applied to a plane or a cylinder or any other shape of feature.

He gives only a hint as to how use can be made of the formalism. It is however, a much more rigorous description of what is actually meant by tolerances than any other work described here and has made possible much of the work described in this thesis. Chapter 4 gives a detailed account of the contents of Requicha's paper and so further discussion will be left till then.

In another paper, Requicha and Chan (1985) describe a constructive solid geometry (CSG) modelling system to which tolerance information has been attached. This is based on the formalism in the previous paper. The goal of this work is to allow process planning to take account of the required tolerances of the part. No attempt is made to analyse the tolerance scheme for functionality or consistency. The tolerance information is stored in a "variational graph". It has nodes representing features and datums. Attributes representing tolerances are attached to each feature.

Each feature is associated with a face of a CSG primitive - the simple geometric solids which appear at the leaves of the CSG tree. Since the same CSG primitive may appear at more than one leaf of the tree care must be taken to give unique specifications of occurrences of primitives.

The paper discusses problems that occur during editing of the CSG model and the variational graph. The two entities must be kept consistent with one another during editing.

Conclusion

A usable system that could deal with tolerances would be based on a sound formalism and would contain a geometric model which included tolerance information. It would allow constraints to be derived from the model and would allow the constraints to be solved to answer questions concerning the toleranced part. However, all the work discussed here tackles this sequence of problems in a limited way.

Bjϕrke formalises dimensional tolerances and their behaviour by forming classifications and shows how results can be obtained. However, he does not include a geometric model of any kind. Requicha describes a very useful formalism of geometric tolerances in his first paper. This is the only work that deals with geometric tolerances. In his second paper, coauthored by Chan, he uses this to attach geometric tolerances to a CSG model. However they do not tackle the problem of deriving results from the model.

The only work that covers all three points mentioned at the end of the introduction to this section is Hillyard. Hillyard describes how results can be obtained from a geometric model with dimensional tolerances. However, because he considers that all edges and vertices are well defined his formalism is limited to dimensional tolerances.

Hoffman shows how results can be obtained from constraints representing the possible variations in positions of features. He does not show how the constraints can be derived and does not include a geometric model or a formalism of tolerances.

Requicha's formalism and generalisation of standard tolerancing practice has formed a basis for the work presented in this thesis. It is has allowed methods for representing toleranced parts and assemblies of toleranced parts to be found. Requicha has focussed on geometric tolerances as opposed to the dimensional tolerances of most of the work described above. The representation is based upon surface features in contrast to edges and vertices as is the case with Hillyard's work.

2.2. RELATED WORK CONCERNING INEQUALITY CONSTRAINTS IN ROBOTICS

Inequality constraints are used to represent incomplete information in robot planning. They can represent uncertainty in the knowledge of the planner when it is generating the plan or uncertainty in the knowledge of the controller when the plan is being executed. The former occurs when certain facts about task parameters are not yet determined. In some cases a definite value must be chosen for these parameters at run-time that satisfies the constraints. However, this is not always the case and so a plan-time uncertainty may also be a run-time uncertainty. Run-time uncertainties are variations in locations and sizes of objects which cannot be determined even at run-time. In this thesis all uncertainty can be classified as run-time uncertainty since it is unavoidable. However, the work described here is also relevant to plan-time uncertainty.

Listed below are sources of run-time uncertainty that must be taken into consideration when planning a robot task.

- The positions at which parts are provided.
- The position of the manipulator.
- Sensor readings.
- The shapes of parts to be manipulated.

The work to be described below shows how knowledge of uncertainty can be derived from the first three of the above sources. However, uncertainty in shape receives little attention. Although it is mentioned by Brooks (1983), it is not treated any differently from position uncertainty.

Some work has assumed knowledge of the size of uncertainties in the planning of a robot task. For example, Lozano-Pérez et al (1984) evaluate fine motion trajectories which can be guaranteed to operate in the presence of uncertainty in position and velocity of the robot. Yin (1984) uses verification vision to determine the position of objects. A "tolerance" is assigned to the position of objects so that bounds can be determined on the position of the image of the object. However, neither of these say how the uncertainties are to be determined.

There are certain topics of particular interest to this thesis which are dealt with to different degrees by the work described below. The topics are:

1. The world must be represented in a way which allows uncertainty information to be attached.
2. Uncertainty must be represented. In all cases described below inequalities are used though the variables have different semantics in different cases.
3. Initial constraints must be obtained from some source. Either they can be assumed to be already provided or they can be derived from analysis of the geometry.
4. Inferences must be performed on initial constraints to determine how they combine and to determine the end results.

Part of Taylor's thesis (1976) deals with the representation and analysis of geometric uncertainty. He uses a representation of geometry which is similar to that used in the object-level programming language, RAPT, discussed in chapter 3. Basically, a part is represented as a collection of features each of which has given size and position relative to some global coordinate system of the part. A pair of features may have a relationship between them to represent the fact that these features are in contact. Such a relationship constrains the positions of the parts.

Taylor discusses two different methods by which uncertainty can be dealt with.

The first method deals with plan-time constraints on the positions of two contacting parts. Constraints are represented as inequalities involving variables representing the positions of the objects. The constraints are determined from knowledge of the sizes of the contacting features. Geometric models are used which consist of collections of surface features whose shape and position are known. A contact can take place between any pair of similarly shaped features. Techniques are provided by which the constraints from multiple contacts can be combined to find their total effect on the positions of the parts.

One disadvantage of this type of representation is that the analysis techniques can only be applied if there are enough constraints to determine the axis of rotation. Therefore, "imperfect contacts" (chapter 3) cannot be dealt with since these may allow limited rotation about any axis. Nor is it obvious how this representation can be extended to deal with other types of uncertainty.

The second method for representing uncertainty, suggested by Taylor, involves differential approximations. The approximations are reasonable only when rotations are small. These are run-time uncertainties. The advantage of this method over the previous one is that the axis of rotation need not be determined. It also introduces the possibility of dealing with kinematic chains of parts. For example, he gives an example of an axle held in a clamp in which both the position of the axle in the clamp and the position of the clamp on the table are uncertain. The system can determine the uncertainty in position of the axle relative to the table.

Unfortunately, he does not go into how such uncertainties can be derived from geometric models. It is assumed that the user of the system knows what the initial uncertainties are. He does not suggest how arbitrarily complex assemblies could be analysed.

Taylor's two representations of position uncertainty are complementary. The first can be used to make an initial analysis involving large uncertainties. When uncertainties have been reduced the second method can be utilised. At the outset uncertainties are often large and in most cases there is only one axis of large uncertain rotation because many contacts allow only one rotation. Therefore, the first representation of uncertainty is suitable. However, most assembly operations require uncertainty to be small before they can be performed. This implies that sensors or other uncertainty reducing operations must be included at some point in the plan. At the point in the plan after uncertainty reduction and before the assembly operation an analysis restricted to small uncertainties is possible. This is where Taylor's second representation can be brought into play.

In conclusion, Taylor describes two ways that bounded locations of toleranced parts can be represented by inequality constraints. Both have a role to play in determining and verifying a plan for robot assembly. Two areas not covered by Taylor are the derivation of small uncertainties from contacting surfaces and the generalisation of kinematics in assemblies of parts. Nor does he deal with parts whose shapes are uncertain.

Brooks (1983) formalises the propagation of uncertainties through robot plans. The types of uncertainty that can be supported by his techniques include uncertainty in location of the robot and the work pieces and uncertainty in sensor readings. A plan consists of a sequence of actions in which there may be incompletely specified parameters representing plan-time uncertainty.

The world is represented by sets of variables and sets of inequality constraints involving these variables. Every action is described as a set of inequalities involving initial variables and final variables. There is a set of constraints that must be satisfied before an action can take place. For example, if an action involves the robot picking a part off a table then the sum of the uncertainty in the positions of the robot and the part must be less than a certain amount. The state of the world after each action must allow the operation of the next action.

An important part of the representation is that every real-world quantity, such as a position or a dimension, has a nominal component and an uncertain component (uncertain at run-time). The sum of these components equals the value of the quantity in the real world. At plan time the nominal part may be unknown. However, it is possible that during planning it will become constrained and so at execution time the nominal part may be known. The uncertain part of any quantity, on the other hand, is always unknown even at execution time.

A sensor is represented as a constraint between the uncertainty in the measurement of the quantity and the nominal value of the quantity. The action of making a sensor reading is represented as an equality between the variables representing the result of the reading and variables representing the physical quantity being measured. Suppose the quantity being measured is represented by nominal component N and an uncertainty component U and that the reading consists of nominal and uncertainty components, N_s and U_s. Then the following constraint holds:

$$N+U = N_s+U_s .$$

Instead of validating whether a plan will succeed or fail an attempt is made to find additional sets of constraints that will ensure that it will succeed. Firstly, constraints involving only nominal variables are sought for. If that is unsuccessful then an attempt is made to find suitable constraints involving the uncertainty variables. In this case there is an additional requirement that there must be a sensor available that could achieve these constraints.

Thus, there are two ways that the system can make suggestions about the plan in order to keep run-time uncertainty at a level which allows operation of the plan.

- Firstly, constraints may be placed on the plan parameters.
- Secondly, appropriate sensor readings may be inserted at appropriate points in the plan.

There is no geometric model in this work. Instead all geometric constraints are assumed to have been reduced to algebraic constraints. There is no appreciation of the geometry of simple actions. For example, if a part is moved from one place to another then the fact that its final position uncertainty is a combination of its initial uncertainty and the robot's position uncertainties must be input as inequalities involving relevant variables. Brooks provides useful formalisations of many areas involving uncertainty in robot plans.

The major analytical tool used to deduce results from constraints is the SUPINF algorithm. This is described in (Brooks 1981). It allows the bounds of a variable constrained by a given set of inequalities to be determined. Since this algorithm plays an important part in this thesis it will be described in more detail in section 3.5 and appendix 2.

Conclusion

Neither Taylor nor Brooks deal specifically with toleranced parts. They deal mainly with uncertainties in position of parts. The approach taken by Taylor is to derive uncertainties partly from geometric models and to determine the build up of these uncertainties and the effect of combinations of constraints. He does not generalise the methods to an assembly of arbitrary complexity. Brooks produces a useful formalism and considers many interesting issues, for example, dealing with uncertainties in sensors.

There are several ways in which the work described here has contributed to this thesis. Most importantly is Requicha's work which will be described extensively in chapter 4. Brooks has provided useful comments on the nature of uncertainty and has described a useful formalism of uncertainties in robot plans. The SUPINF algorithm has been of direct relevance to this thesis. Other work provides interesting comparisons with this thesis in the areas covered and the methods by which they have been achieved.

A subject not considered by any of the work described in this chapter concerns imperfect surfaces in contact. The only work which says anything about uncertainties between contacting parts is (1) Bjϕrke one of whose dimension classifications concerns dimensions between parts and (2) Taylor who derives constraints from perfect contacts between perfect features. Nor has very much been achieved regarding the representation and analysis of a geometrically toleranced part in a computer.

Chapter 3: ASSEMBLIES OF NOMINAL PARTS

Introduction

This chapter shows how a structure of poorly fitting but perfectly formed (nominal) parts can be analysed. The exact position of each part is uncertain because they can slop about in the assembly. It is assumed, however, that parts can only deviate from their nominal positions by a small amount. The parts are prevented from making larger deviations by the contacts that occur between them (Fleming 1985a and 1985b).

The following assumptions are made about parts and assemblies of parts.

- All parts are rigid.
- Parts are not glued or bolted to one another in any way.
- The nominal position of each part is known.
- The parts are not subject to any force such as gravity or tightened bolts.

Note that forces constrain the positions of parts. For example, if a block is above a table with nothing between the two objects then gravity would constrain the block to be in contact with the table. Such a constraint cannot be inferred if gravity is ignored.

The surfaces of parts are to be divided into features which are simple geometric entities such as planes, cylinders and spheres etc. The size of features is assumed to be much larger than the amount that parts can deviate from their nominal positions.

There is a set of pairs of features between which potential contacts can occur. This implies that the features can come into contact or can be separated. It is assumed that all pairs of potentially contacting features are known.

In the first section of this chapter the off-line robot programming language, RAPT, is introduced. RAPT includes a powerful geometric reasoning system which forms the basis for some of the work described in this thesis. As it stands RAPT has no concept of geometric uncertainty but the representation of nominal parts in the uncertainty reasoning system, to be described here, closely follows the representation used by RAPT. Section 2 discusses improvements to RAPT's representation of geometry necessary for supporting analysis of uncertainty. Section 3 shows how uncertain positions can be represented by inequality constraints. Each contact between parts gives rise to constraints on their positions. In section 4 it will be shown how these constraints can be evaluated from the shape of the features in contact. Ultimately the positions of the parts are determined by the combined effect of all of these constraints and section 6 will show how this can be achieved.

3.1. INTRODUCTION TO RAPT

In this thesis the representation of nominal geometry is based on the representation used by the object-level robot programming system, RAPT. RAPT has been developed in the Department of Artificial Intelligence at the University of Edinburgh (Ambler et al 1983) and (Popplestone and Ambler 1981).

RAPT includes a powerful geometric reasoning system. It allows the shapes of parts to be defined and a program for their assembly to be described. The required contacts between the parts are described by "geometric relationships" between features of the parts. A motion of the robot can be described by stating the relationships that the motion should attain.

Features

A part is a rigid solid object whose surface is represented as a collection of features. A feature is a simple geometric surface such as a plane, a cylinder or a sphere. Planes and cylinders are unbounded and have infinite extent. Points and lines are represented as spheres and cylinders of zero radius.

Each feature has a coordinate system attached to it and the position of that coordinate system relative to the main coordinate system of the part is given. Feature coordinate systems are attached in standard configurations. For example, the coordinate system of a plane has its x-axis normal to the plane and its origin lying in the plane. The coordinate system of a cylinder has its x-axis parallel to the cylinder's axis and its origin lying on the axis.

Each feature and part is given a name. A feature can be specified by giving the names of the feature and the part to which it belongs.

There are various convenient ways that the positions of features can be specified. For example a plane may be stated as being parallel to another feature and a certain distance from it or may be made to pass through three given points. The description of a part can be incomplete since only the features important to the manipulation plan need be defined.

Geometric Relationships

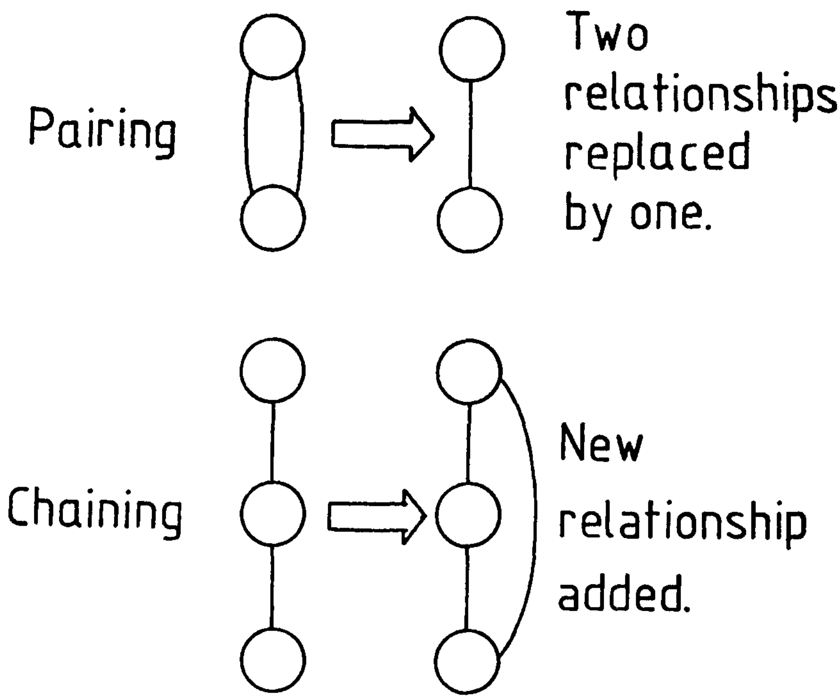
Geometric relationships are used, in RAPT, to describe how parts interact. A feature of one part may fit into or be against a feature of another part. For example a box lying on a table has its bottom surface against the top of the table. This condition is described by asserting an AGAINST-relationship between the feature representing the bottom of the box and the feature representing the top of the table. Similarly, if a cylindrical peg is inserted into a cylindrical hole then a FITS-relationship is said to hold between the features representing the peg and the hole.

A relationship between two parts removes some of their relative degrees of freedom. A completely free part has six degrees of freedom which can be taken to be three rotations and three translations. However if two parts have a geometric relationship between them then some of their relative degrees of freedom are removed. For example if an AGAINST-relationship holds between two planar features then only three degrees of freedom are possible: two translations and one rotation. Motion can take place in these degrees of freedom while still preserving the relationship. A FITS-relationship between two cylindrical features constrains their axes to be colinear. There are, then, two possible degrees of freedom: one rotation and one translation.

There are other types of relationship which are not usually present in the initial model of the assembly but arise as a result of the inferences described in the next section. One of these is the FIX-relationship which allows no degrees of freedom.

Reasoning

RAPT can perform inferences over geometric relationships to find further constraints on the positions of the parts. If sufficient relationships have been provided then eventually the nominal locations of all parts will be deduced. The two inferences are illustrated in figure 3.1.1.

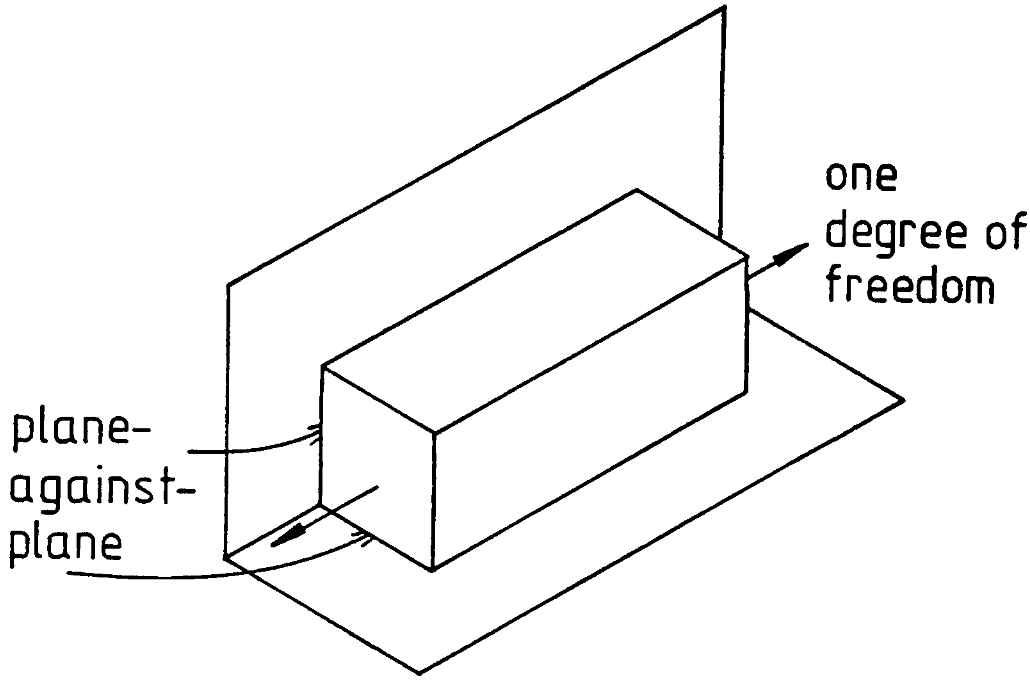


The two inferences performed by RAPT. The first replaces a pair of parallel arcs by a single arc. The second introduces a new arc which has the same effect as two arcs in series.

Figure 3.1.1

Each relationship applies between two parts and constrains their relative location. The parts can be represented by the nodes of a network and the relationships by arcs. The network is analysed to find loops which will be called "cycles". The simplest cycle involves just two parts and two relationships and is referred to as a 2-cycle.

The first type of inference, called "pairing", replaces the two relationships in a 2-cycle by a single equivalent relationship. New features are generated and the new relationship acts between these. For example two plane-against-plane relationships may exist between two parts with perpendicular planes. There is one linear degree of freedom possible: the parts may slide parallel to the edge at the intersection of the planes (figure 3.1.2). The system creates two new features to represent the edges at the intersections of the planes. A relationship is set up between them called a "LIN-relationship". It states that the x-axes of the coordinate systems of the new features are colinear and their y- and z-axes are parallel.



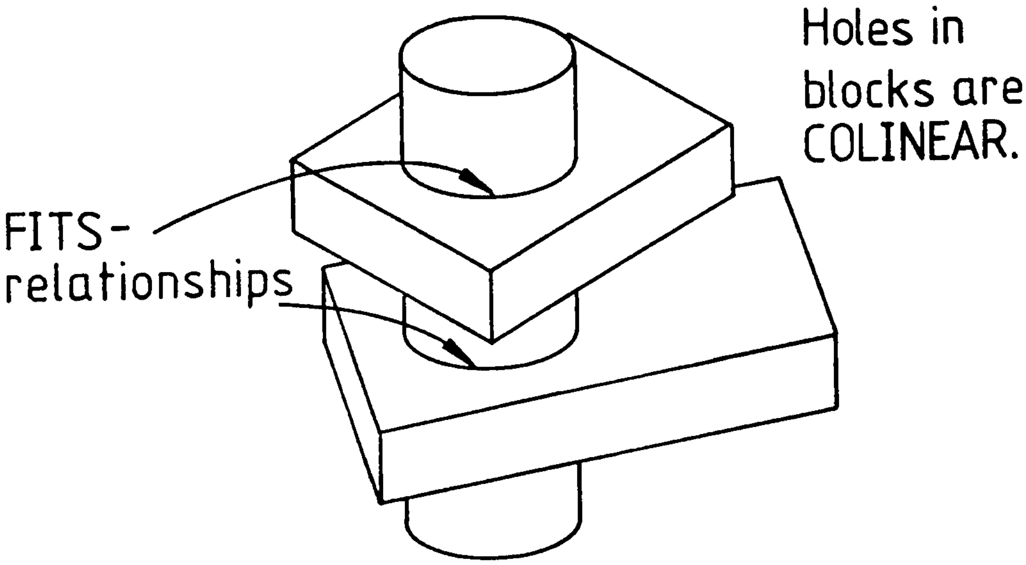
Two non-parallel plane-against-plane relationships allow a single degree of freedom.

Figure 3.1.2

If the planes on these parts were parallel instead of perpendicular then the two relationships together would be no more constraining than the relationships individually. In this case one of the relationships is deleted. If it is physically impossible to make both relationships simultaneously then an error is reported.

When all 2-cycles have been removed by replacing pairs of relationships by a single equivalent relationship then larger cycles are searched for. They are dealt with using the following inference, called "chaining".

Suppose there is a relationship between parts A and B and a relationship between parts B and C. Then in some instances a relationship can be deduced between A and C. For example consider two parts with a hole in each and a peg inserted through both holes, (figure 3.1.3). The peg forms a FITS-relationship with each of the parts. The result is that the holes in the two parts are constrained to be colinear. There is a rule which says that, in general, we can deduce a colinear relationship between two parts whenever they are linked by a third part which forms colinear FITS-relationships with them.



Two parallel FITS-relationships creating a colinear relationship.

Figure 3.1.3

Another example is two rectangular blocks one on top of the other on a flat table. The bottom of the top block is constrained to be parallel with the table and at a fixed distance from it. An AGAINST-relationship is formed between the table top and an imaginary surface at a fixed distance above the table.

A 3-cycle can be dealt with using the two inferences as follows. Firstly, chaining is used to produce a new relationship between two of the parts. There is now a 2-cycle which can be reduced by application of pairing. The resulting relationship expresses the combined effect of all three relationships in the original 3-cycle.

RAPT has a look-up table for each of the two inference methods. Each table has an entry for every pair of relationship types to which the inference can be applied. The table indicates the type of the new relationship and explains how to produce the new features between which the relationship holds. Also included are certain conditions which the input relationships must satisfy. For example, some inferences can only be made if the features involved are parallel.

Limitations of RAPT

There are four major deficiencies in the current RAPT system. Firstly, descriptions of parts in RAPT are incomplete in two ways. All features (excluding spheres and points) are assumed to be infinite and so they have no defined boundary.

The second deficiency of RAPT is that descriptions of parts are often incomplete because only a subset of their features are specified. This is because specifying all features is unnecessary and would create redundant constraints.

Thirdly, RAPT assumes that the world is perfect. Whenever two parts are fitted together there will be a small amount of movement in all degrees of freedom (unless the parts are somehow bound together). However, RAPT assumes that some degrees of freedom are removed completely.

Fourthly, RAPT assumes that parts have ideal geometric shape. No account is taken of the fact that the shape of a real part is imperfect and is infinitely complex and that there is infinite variation between different copies of the same part.

Hence RAPT cannot deal with geometric uncertainties in any way. Before uncertainty analysis can be added to RAPT it would be necessary to enhance the representation of parts.

3.2. NOMINAL GEOMETRY

The representation of toleranced parts consists of two components: the nominal description and the tolerance specification. This scheme is followed by both Hillyard (1978) and Requicha (1983a and 1983b). The representation of the tolerance specification will be left till chapters 4 and 5. The nominal description will be described in this section.

The representation of parts closely follows the representation used in RAPT in that the surface of a part is divided into features which are simple geometric entities such as planes and cylindrical surfaces. Line-features and point-features will not be dealt with. The position of each feature is known with respect to the main coordinate system of the part.

The main difference from representation in RAPT is that all features are finite. Each feature has an extent which refers to the area that its surface covers. The extent of a feature is determined by its boundary.

Some features have the property of size. In the case of cylinders and spheres this refers to their diameter. Planar features, however, do not have size. The area covered by a planar feature is referred to as its extent.

The parts must be described to the system in greater detail than is in general required by RAPT. Relationships can be redundant in RAPT's nominal analysis but important in an uncertainty analysis. For example consider a rod which fits through two colinear holes. In RAPT one of the holes could be omitted from the description because both relationships imply the same constraints on the rod. However, in analysing uncertainties we are interested in the possible variation of the inclination of the rod. This variation is greatly reduced when there are two holes instead of one hole. The description of geometry in a system for analysing uncertainties must include all features which can make contact with other features.

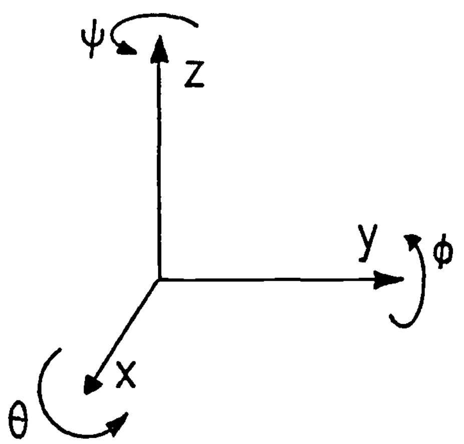
An assumption to be made about assemblies of parts is that the parts are not fixed to one another in any way. Their positions are constrained only by the contacts between them.

The concept of a <u>potential</u> <u>contact</u> is introduced. A potential contact between two features implies that the pair of features may come into contact or that they may be separated. Each potential contact puts constraints on the positions of the parts involved because the features cannot penetrate one another. A potential contact is equivalent to a geometric relationship in RAPT though the constraints it implies are weaker. This is because a RAPT relationship removes some degrees of freedom altogether whereas a potential contact merely puts restrictions on them.

3.3. REPRESENTATION OF POSITIONS

Each part has a <u>nominal</u> <u>position</u>. This is an abstract concept introduced because it is useful to have an ideal position for parts from which their deviations can be measured. When given the nominal shapes of parts and the relationships between them RAPT will infer the nominal positions of the parts. They are specified relative to some world coordinate system.

A position can be defined with six values representing three translations and three rotations. They will be denoted x, y, z, θ, ϕ and ψ where x, y and z represent the position of the part's origin in space and θ, ϕ and ψ represent its orientation. The relationship between axes of rotation and directions of translation is shown using the right-hand coordinate system in figure 3.3.1.



The axes about which rotations θ, ϕ and ψ are taken.

Figure 3.3.1

3.3.1. Position Uncertainties

The position of a part is subject to uncertainty. An uncertainty in a position is defined to be the set of possible deviations of the position from its nominal value: it can be thought of as a set of values that the position can take. This section will describe the representation of constraints used to define this set.

Suppose a part can undergo small variations from its nominal position. A coordinate system in a part can be used to define six degrees of freedom by taking three translations parallel to the axes and three rotations about the axes. Small variations from the nominal position will be expressed in terms of variables referred to as "degree of freedom (DOF) variables". They will be denoted by Δx, Δy, Δz, $\Delta\theta$, $\Delta\phi$ and $\Delta\psi$, the Δ prefix being used to indicate that these are small variations.

Constraints may be represented as inequalities involving these variables. Of course the precise form of the inequalities depends on the choice of coordinate system. We shall say that a set of inequalities represents an uncertainty with respect to a particular coordinate system.

There will be many coordinate systems in each part. When a set of constraints is initially generated a coordinate system must be chosen with respect to which to represent it. Later, a different coordinate system may be found to be preferable and the inequalities need to be converted so that they represent the same uncertainty but with respect to a different coordinate system. Two representations are equivalent if they define the same set of possible positions for the part.

The DOF-variables of the old coordinate system can be expressed in terms of the DOF-variables of the new coordinate system. The inequalities are converted simply by substituting each occurrence of an old DOF-variable in the inequalities by an equivalent expression involving the new DOF-variables.

It is useful to think of position uncertainties in terms of configuration space. This is a term used by Lozano-Pérez (1979) in discussing collision avoidance algorithms. A point in a part's configuration space corresponds to the position and orientation of the part. It follows that the set of possible positions specified by an uncertainty can be represented by a region of configuration space. For convenience the origin of configuration space of a part will be taken to be the nominal position, for that part, output by RAPT.

Often there will be inequalities, for each DOF-variable, ΔD, of the form,

$$\text{expression1} \leq \Delta D \leq \text{expression2},$$

where the expression1 and expression2 do not involve ΔD but may involve other DOF-variables. The two expressions will be referred to as the bounds of ΔD.

In general the expressions can be of arbitrary complexity. They may be non-linear and involve trigonometric functions. However, it is often useful to make simplifications to approximate the bounding expressions to allow them to be analysed more easily. This introduces inaccuracy. The following section discusses approximations and their validity.

3.3.2. Approximations

Since the expressions involved in inequality constraints may become complex it is useful to make simplifications so that the inequalities are easier to analyse. However simplifications inevitably mean that the inequalities only approximately represent the region of configuration space that they are intended to represent. Nonetheless, there are two types of simplification which can be made that still allow useful results to be obtained.

Firstly, some simplifications made to the inequalities have effects which are small enough to be ignored. For example, under the assumption that all rotations are small, trigonometric functions can be linearised $(\sin(\Delta\theta)=\Delta\theta \ , \ \cos(\Delta\theta)=1)$.

Secondly, there are simplifications to the inequalities which cause larger changes to the region of configuration space. Sometimes it is possible to tolerate a set of inequalities which overestimates the actual region of configuration space and sometimes it is possible to tolerate an underestimate. Here, an overestimate of a region of configuration space is defined to be a region **containing** the actual region and an underestimate is defined to be a region **contained** in the actual region. Whether it is an overestimate or an underestimate that can be tolerated depends on the results that are ultimately required. Sometimes we require to find an upper bound to some value and then an overestimate is satisfactory and sometimes we requires a lower bound and then an underestimate is satisfactory.

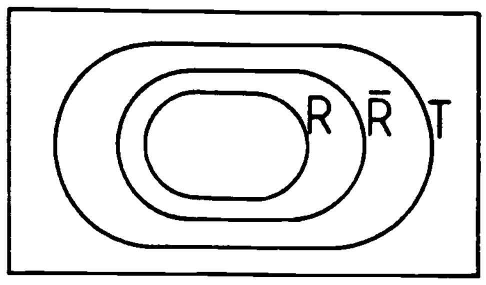
Put another way, some properties of an actual region of configuration space can be verified by testing the property on an overestimate of the actual region whilst others can be verified on an underestimate.

As an example of the first case suppose we require to test that some actual region, R, is contained in some set T. Let $\bar{R}$ be an overestimate of R. Then if $\bar{R}$ is found to be contained in T then we know that R is also contained in T. Figure 3.3.2 is a Venn-diagram illustrating this.

For the second case suppose we require to test that R is non-empty. Let $\underline{R}$ be an underestimate of R. Then if $\underline{R}$ is found to be non-empty then we know that R is also non-empty.

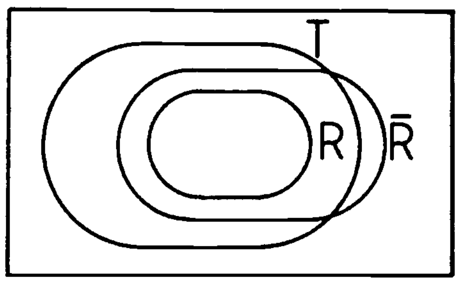
Note that in both cases we cannot infer the opposite case. We cannot infer, for example, that R is not contained in T because $\bar{R}$ is not contained in T. Figure 3.3.3 shows R contained in T but $\bar{R}$ not contained in T.

The two properties mentioned above are useful for analysing a structure of parts. If the region of configuration space for a part is contained in a set T then it implies that the parts are held with a tightness represented by the size of T. T can be



If an over-estimate of R is contained in T then R must be contained in T.

Figure 3.3.2



If an overestimate of R is not contained in T then we cannot deduce that R is not contained in T.

Figure 3.3.3

chosen to reflect the required tightness of the parts. The second property that, a region of configuration space is non-empty, is useful for testing that there exist valid positions for a part. If the configuration space for a part is empty then there are no possible positions for the part implying that the parts cannot be fitted together.

In practice the above considerations allow all inequalities to be linearised in DOF-variables. Rotational terms can be linearised by keeping all rotations small. This allows trigonometric approximations to be made: $\sin(\Delta\theta) = \Delta\theta$ and $\cos(\Delta\theta) = 1$. These can be put under the category of small approximations. Larger approximations arise when linearising inequalities associated with contacts between curved surfaces and when the area of overlap of features is approximated. In these cases it is possible to arrange that larger approximations are overestimates or underestimates.

3.4. DERIVATION OF CONSTRAINTS FROM RELATIONSHIPS

In RAPT a relationship between two parts removes some of the degrees of freedom of the parts. In the presence of uncertainty, however, no degrees of freedom are removed completely though they are constrained by the relationship.

A potential contact between two features constrains the positions of their associated parts. Each feature has a material side and a non-material side and the material sides do not intersect. This is a geometric constraint. It can be converted to an algebraic constraint involving DOF-variables by considering the "signed distance" between the two features. A signed distance is defined as follows.

1. When the material sides of the features do not interfere it equals the distance between the closest points of the features;
2. When the material sides do interfere it is **negative** and represents the shortest distance that one of the parts could be moved to remove interference.
3. When the features are in contact it equals zero.

Despite the three part definition the signed distance is a continuous function of the relative position of the parts.

Example 1: Two-Dimensional Peg in Hole

Consider figure 3.4.1 showing a peg of diameter, Dpeg, loosely fitting in a hole of diameter, Dhole. For simplicity, this example is two-dimensional. A coordinate system is positioned at the midpoint of the range of overlap. There is only one rotation denoted by θ. It acts about an axis through the origin of the coordinate system perpendicular to the page. Translation along the axis of the peg is unconstrained.

The range of overlap of the peg and the hole is illustrated and has length, E. The difference between the diameters of the hole and the peg, Dhole-Dpeg, will be denoted by Ddiff. The signed distance between the features, assuming linearisation of the rotations is,

$$\mathrm{Ddiff} + \min\left(\Delta y+(E/2)\Delta\theta,\ -\Delta y+(E/2)\Delta\theta,\ \Delta y-(E/2)\Delta\theta,\ -\Delta y-(E/2)\Delta\theta\right).$$

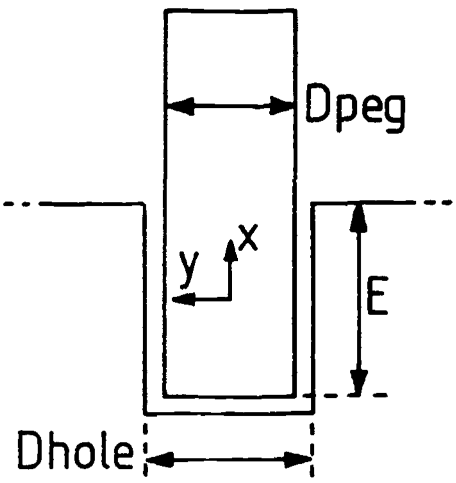
Therefore the constraints on their positions can be written,

$$\mathrm{Ddiff} \pm \Delta y \pm (E/2)\Delta\theta \geq 0,$$

which represents four inequalities with different combinations of plus and minus signs. Note that these inequalities represent the required region of configuration space exactly except for the linearisation of rotations which fall in the category of small approximations.

Example 2: Three-Dimensional Cylindrical Peg in Hole

The constraints arising from a three-dimensional peg in hole (figure 3.4.2) are similar to those in the two-dimensional case except that there are more degrees of freedom. There are two constrained rotations, ϕ and ψ, about axes perpendicular to the cylinders' axes and two constrained translations, y and z along



A peg loosely fitting in a hole in two dimensions.

Figure 3.4.1

axes perpendicular to the cylinders' axes. Again, the peg and hole overlap by an amount E and the difference between the diameters of the peg and the hole will be denoted by Ddiff.

The constraint that the signed distance between the features is greater than or equal to zero can be expressed as,

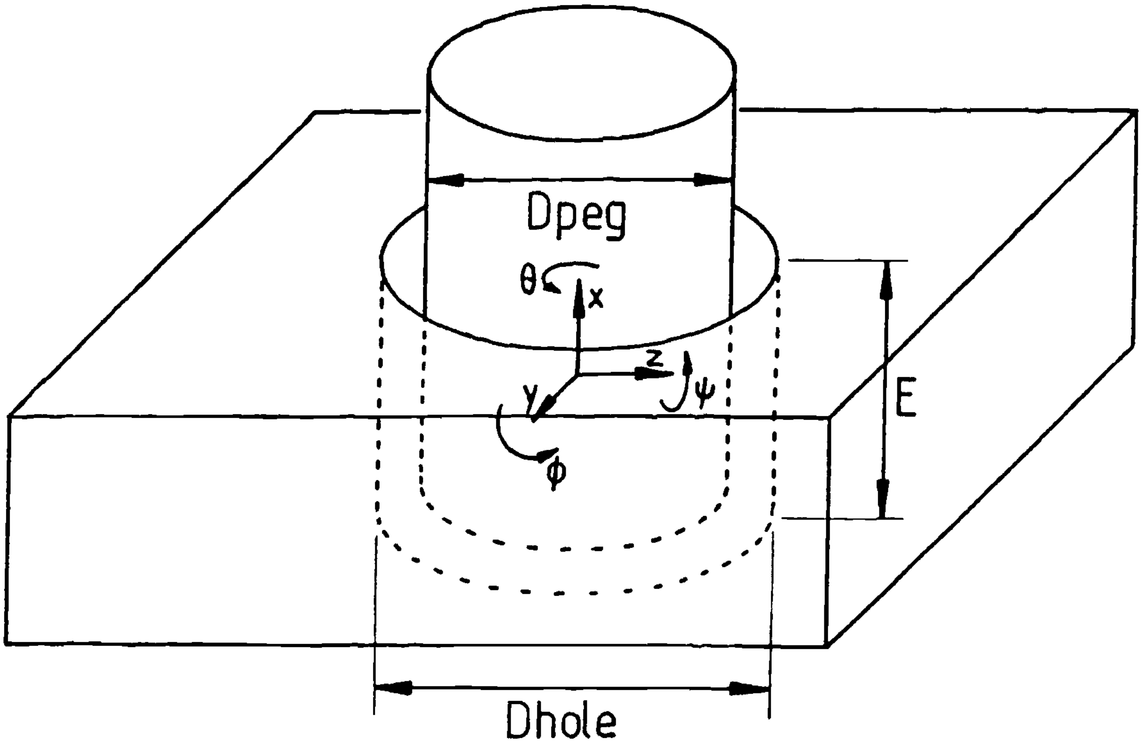
$$\mathrm{Ddiff} - \min_{\alpha=\pm 1}\left(\sqrt{(\Delta y+\alpha(E/2)\Delta\psi)+(\Delta z+\alpha(E/2)\Delta\phi)} \right) \geq 0 \qquad \text{C1}$$

where the terms with α equal to ± 1 correspond to the two ends of the overlap between the peg and the hole.

C1 accurately describes the region of configuration space (except for the linearisation of trigonometric functions). However, this expression is nonlinear and can only be made linear at the expense of introducing approximation. Constraints on Δy and Δz can be separated to produce inequalities of the form

$$-A \leq \Delta y \pm (E/2)\Delta\psi \leq A,$$
$$-A \leq \Delta z \pm (E/2)\Delta\phi \leq A. \qquad \text{C2}$$

The approximation can be made an overestimate or an underestimate



A peg loosely fitting in a hole in three dimensions.

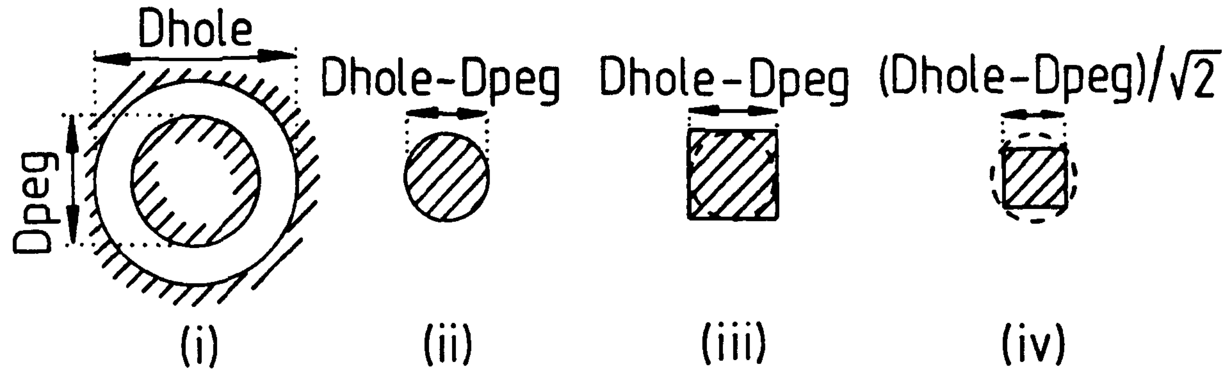
Figure 3.4.2

by appropriate choice of A. Figure 3.4.3(i) shows a cross-section of the peg and hole and figure 3.4.3(ii) shows the associated region of configuration space. Figures 3.4.3(iii) and 3.4.3(iv) show possible over and underestimates of the actual region which could be obtained by constraints of the form C2. The overestimate is obtained by letting

$$A=(Dhole-Dpeg)/2$$

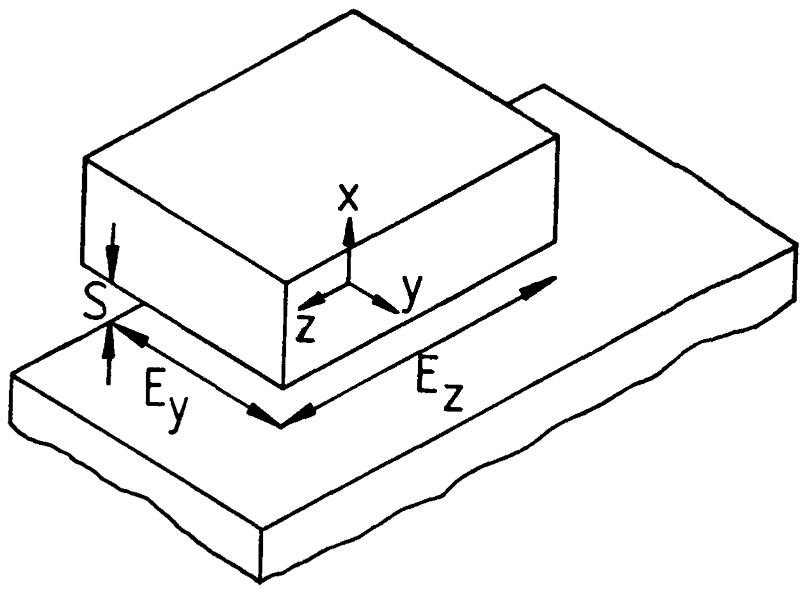
and the underestimate by letting

$$A=(Dhole-Dpeg)/(2\sqrt{2}).$$



(i) A peg in a hole.
(ii) Configuration space of the peg relative to the hole.
(iii) An overestimate of the configuration space.
(iv) An underestimate of the configuration space.

Figure 3.4.3



A potential contact between two planes with a nominal separation of S and overlapping on a rectangle E_y by E_z.

Figure 3.4.4

Example 3: Potentially Contacting Planes

As another example consider a plane-against-plane relationship shown in figure 3.4.4. The area of overlap of the features is assumed to be a rectangle with dimensions E_y and E_z as illustrated and the features have a nominal separation of S. When the area of overlap is not a rectangle it can be approximated by enclosing it in a rectangle or by fitting a rectangle inside it. In the former case tighter constraints will be obtained causing an underestimate for the set of possible positions. In the latter case an overestimate will be obtained. A coordinate system is chosen at the midpoint of the rectangle.

The signed distance between the features being greater than or equal to zero implies that, with linear approximations,

$$S + \Delta x \pm (E_y/2)\Delta\psi \pm (E_z/2)\Delta\phi \geq 0.$$

Although the relationship itself does not restrict rotations to be small it can be assumed that there are other relationships which do so. For example, consider a tab fitting into a slot. There are two potential contacts between planes and together they force rotations to be small as long as the difference between the widths of the tab and slot is a lot smaller than the extent of their overlap.

3.5. SUP AND INF

Before going on to describe how constraints can be manipulated, two procedures SUP and INF will be introduced. These can be used for obtaining useful results about how a set of inequalities puts bounds on variables. Algorithms for SUP and INF are taken directly from Brooks (1981) and are described in more detail in appendix 2.

Let A be an expression, V a set of variables and C a set of inequality constraints. Then:

$$SUP(A, C, V)$$

returns an expression involving only variables in V which is an upper bound for A assuming that constraints C hold. If V is empty then a number is returned. If the inequalities in C are linear then the result can be guaranteed to be a least upper bound for A.

For example suppose

$$C = \{ w \le y \ , \ w \le x \ , \ x \le z \ , \ z \le 2 \}.$$

Then the following results would be evaluated

$SUP(w, C, \{w\}) = w$, (SUP(A,C,V)=A if A is a variable and if V contains A)

$SUP(w, C, \{x,y,z\}) = \min(x,y)$,

$SUP(w, C, \{y,z\}) = \min(y,z)$,

$SUP(w, C, \{y\}) = \min(y,2)$,

$SUP(w, C, \{\}) = 2$.

If the inequality "$z \le 2$" had been omitted from C then $SUP(w, C, \{\})$ would evaluate to ∞ since no numeric bound could then be found.

Similarly,

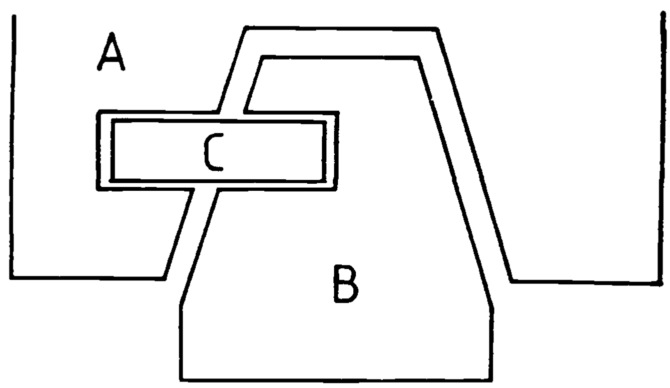
$$INF(A, C, V),$$

returns a lower bound expression for expression A subject to C and which involves variables V. A greatest lower bound is obtained if the inequalities are linear.

3.6. COMBINING CONSTRAINTS

Suppose that each relationship in an assembly has been analysed to find the constraints that it implies on the relative positions of the parts. A network has been generated with nodes representing parts. For each relationship there is an arc carrying the constraints generated from that relationship. These constraints can be combined to determine the constraints on the assembly.

Suppose we want to find the constraints on the relative position of two parts, A and B, of an assembly. There may be arcs from A to B and the constraints from these act on A and B directly. There may also be paths from A to B via other parts. Some of these paths add further constraints though others will not. For example, in figure 3.6.1, parts A and B are in contact but are also linked by part C. Constraints are generated from their direct contact as well as from their connection via C.



Parts A and B are in contact but are also linked by part C.

Figure 3.6.1

There are two types of inference which are made on the network of constraints. They will be described in detail in the following paragraphs. They correspond to the two types of inference, pairing and chaining, made by RAPT to find the nominal positions of parts described in section 3.1 under "Reasoning" and illustrated in figure 3.1.1.

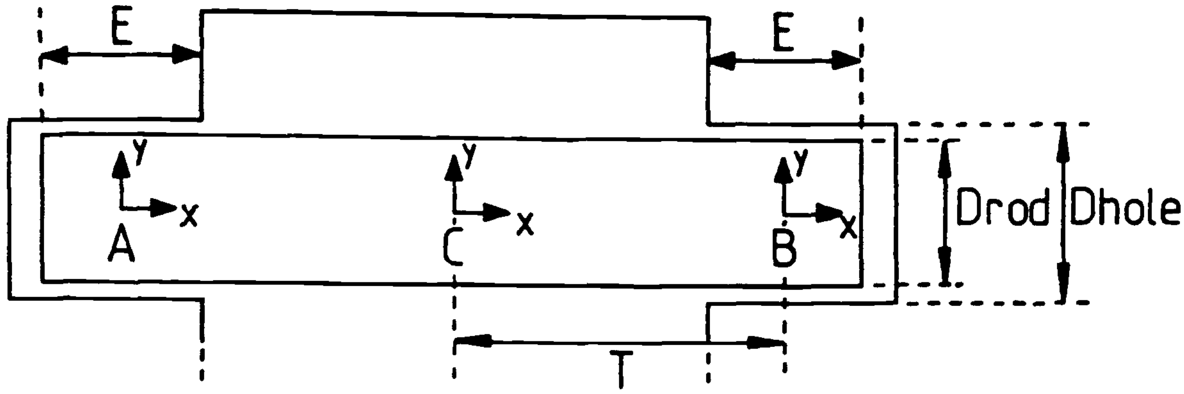
An algorithm is described for deciding the order to apply the inferences. This will also be applicable to the networks discussed in following chapters.

3.6.1. Intersecting Constraints

Two parts may have two sets of constraints acting on their relative positions. The first type of inference finds a set of constraints that expresses the combined effect of both of them. This inference corresponds to RAPT's inference called pairing. It would simply be a matter of taking the union of the constraint sets were it not for the fact that the constraints apply at different features of the parts. The sets of inequalities contain different DOF-variables since each feature has its own coordinate system.

It was mentioned in 3.3.1 how a set of inequalities can be converted to represent the same uncertainty with respect to a different coordinate system. All the sets of constraints to be combined must be converted to a common coordinate system. This coordinate system can be chosen arbitrarily. The required set of constraints is, then, simply the union of all the constraints represented in the common coordinate system.

Note that the constraint sets input to the inference may all be overestimates or may all be underestimates but they may not be a mixture of over and underestimates. The output constraint set will be an over or underestimate according to the input constraints. The inference itself, however, is accurate and so if all inputs accurately represent their regions of configuration space then the output will also be accurate.



A rod held by two bearings. If constraints are derived at each bearing relative to coordinate systems at A and B what are the combined constraints relative to coordinate systems at C?

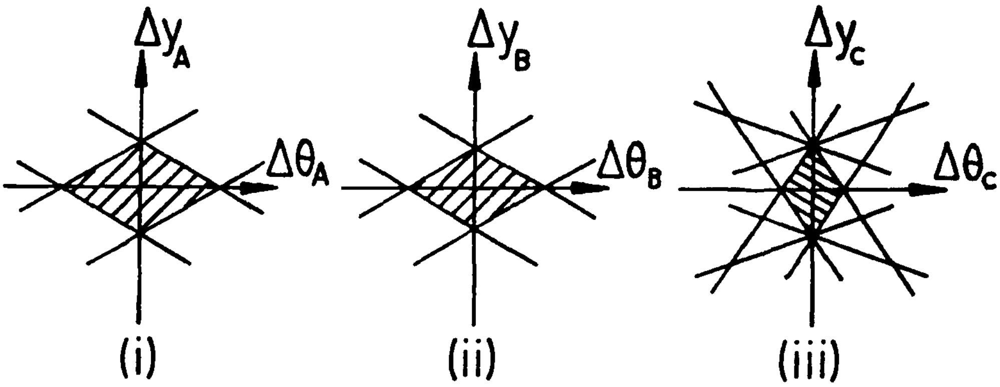
Figure 3.6.2

As an example to illustrate the process, consider the rod held by two bearings shown in figure 3.6.2. For simplicity, the example will be taken in only two dimensions. Each end of the rod fits into a hole in the frame. Constraints are derived from each of these relationships. The letter A, in the diagram, shows the position of two coincident coordinate systems associated with the left hand relationship. They remain coincident only while the parts are in their nominal positions: one coordinate system is fixed relative to the rod and the other is fixed relative to the frame. Similarly there are two coincident coordinate systems associated with the right hand relationship, marked B.

From the left-hand relationship constraints are generated relative to coordinate systems at A. Let Drod and Dhole be the diameters of the rod and hole respectively and let Ddiff be the difference between the diameters, Dhole-Drod. Let E be the distance that the rod extends into the hole (see figure 3.6.2). The constraints obtained are then (note ± signs),

$$-Ddiff/2 \pm (E/2)\Delta\theta \leq \Delta y \leq Ddiff/2 \pm (E/2)\Delta\theta, \quad \text{C1}$$
$$-Ddiff/E \leq \Delta\theta \leq Ddiff/E.$$

The right-hand relationship also gives rise to similar constraints relative to coordinate system B. Figures 3.6.3(i) and (ii) show the regions of configuration space described by these constraints.



(i) and (ii) show the regions of configuration space obtained at the individual relationships in figure 3.6.2 relative to their respective coordinate systems.
(iii) shows these relative to coordinate system C.

Figure 3.6.3

A coordinate system, C, is chosen half-way along the rod as shown in the diagram. It is separated from A and from B by a distance T. Again, this is actually two coincident coordinate systems, one attached to the rod and one to the frame. Constraints C1 are converted to be relative to C. The result is,

$$-Ddiff/2 + (T \pm E/2)\Delta\theta \leq \Delta y \leq Ddiff/2 + (T \pm E/2)\Delta\theta,$$
$$-Ddiff/E \leq \Delta\theta \leq Ddiff/E, \qquad C2$$

where T is the separation of coordinate systems A and C equal to the separation of coordinate systems B and C.

Constraints relative to C which express the effect of the right-hand relationship are identical to C2 except that T has a minus sign. This is because the translation from A to C is the negative of the translation from B to C:

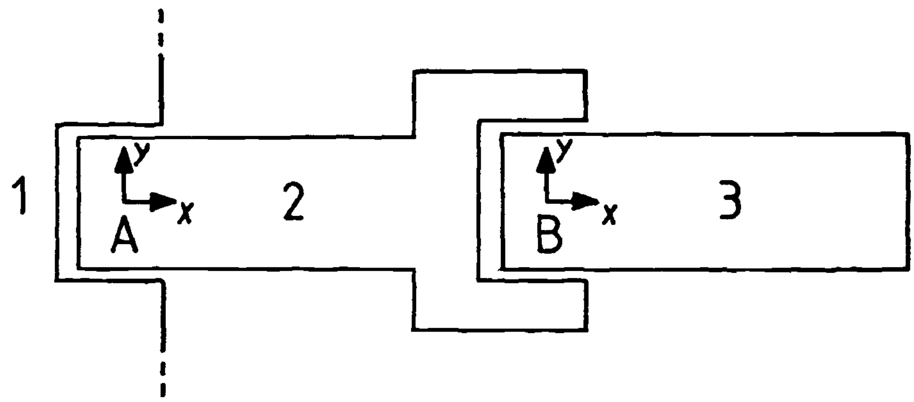
$$-Ddiff/2 + (-T \pm E/2)\Delta\theta \leq \Delta y \leq Ddiff/2 - (-T \pm E/2)\Delta\theta, \qquad C3$$

$$-Ddiff/E \leq \Delta\theta \leq Ddiff/E.$$

The combined effect of the two relationships can be expressed by the union of constraint sets C2 and C3. Figure 3.6.3(iii) shows the regions of configuration space obtained at coordinate system C and their intersection.

3.6.2. Summing Constraints

Uncertainties tend to build up along a kinematic chain. In figure 3.6.4 part 3 is subject to uncertainty caused by its relationship to part 2 and the relationship between parts 2 and 1. What constraints can be found on part 3 relative to part 1? This inference is equivalent to the "chaining" inference of RAPT.



What are the constraints on part 3 relative to part 1?

Figure 3.6.4

There are relationships between parts 1 and 2 and between parts 2 and 3. Coordinate system A is centred on the range of overlap of parts 1 and 2 and coordinate system B is centred on the range of overlap of parts 2 and 3. Only the degrees of freedom Δy and $\Delta\theta$ will be considered in this example.

Introduce the notation $C_{pq}(S)$ to be the region of configuration space of part p relative to part q with respect to coordinate system S.

We know $C_{32}(B)$ and $C_{21}(A)$ and want to find $C_{31}(B)$ or $C_{31}(A)$. It is simplest to find $C_{31}(B)$ and this can afterwards be converted to $C_{31}(A)$ by suitable change of variables. Initially, constraints are derived relative to the coordinate system associated with the relationship (figures 3.6.5(i) and (ii)). However, they must both be expressed in terms of the same coordinate system before they can be combined. Figure 3.6.5(iii) shows $C_{21}(B)$, the region of configuration space of the first relationship relative to coordinate system B.

The DOF-variables representing the position of part, p, relative to part, q, (with respect to coordinate system B) will be denoted by,

$$\Delta x_{pq},\ \Delta y_{pq},\ \ldots,\ \Delta\psi_{pq}$$

and $\mathbf{p}_{pq}$ will denote the vector,

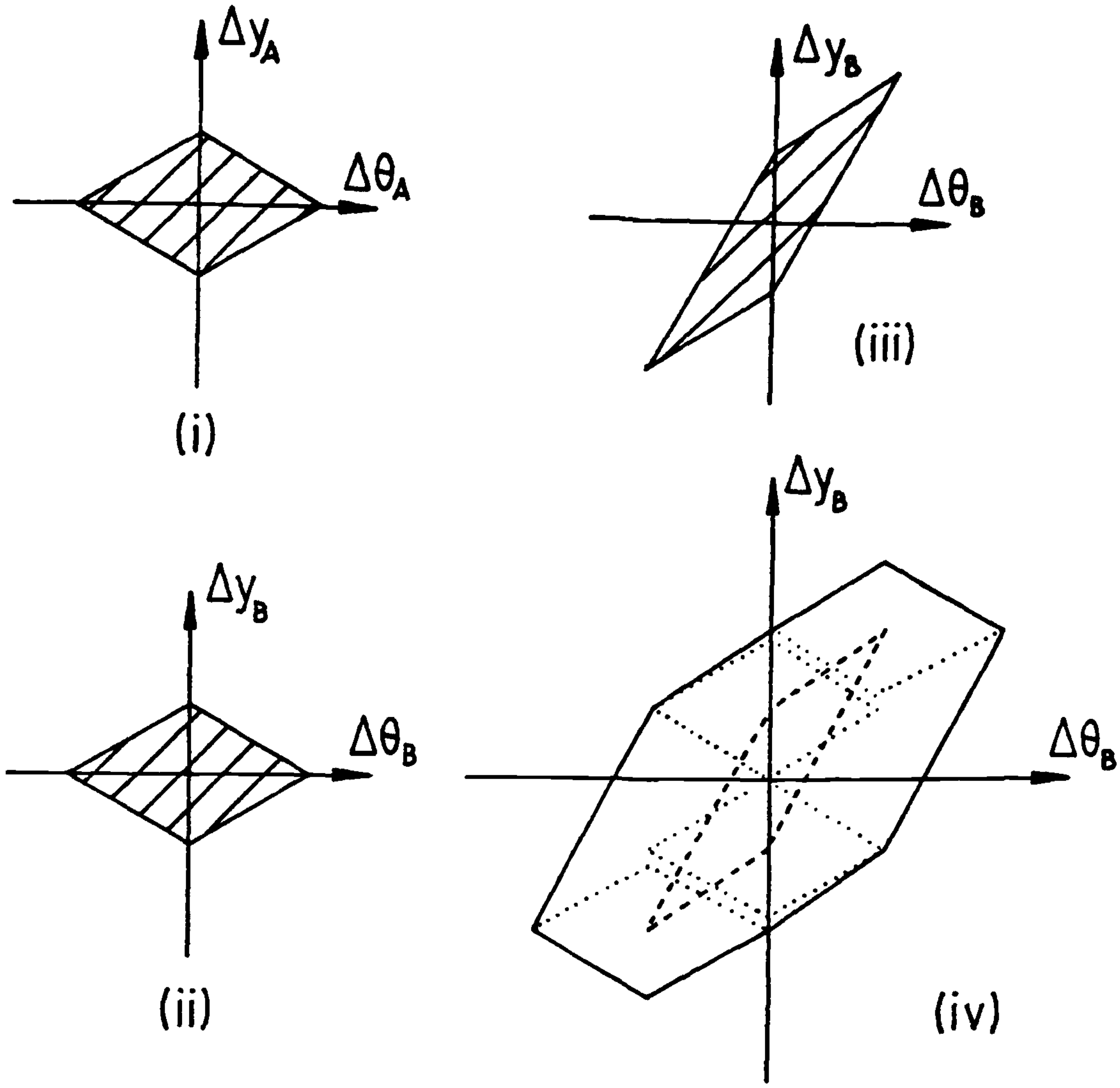
$$(\Delta x_{pq},\ \Delta y_{pq},\ \Delta z_{pq},\ \Delta\theta_{pq},\ \Delta\phi_{pq},\ \Delta\psi_{pq}).$$

Then the small uncertainty assumption means that these vectors can be added together in the following way:

$$\mathbf{p}_{31} = \mathbf{p}_{32} + \mathbf{p}_{21}.$$

In other words for ΔD in $\{\Delta x,\ \Delta y,\ \Delta z,\ \Delta\theta,\ \Delta\phi,\ \Delta\psi\}$

$$\Delta D_{31} = \Delta D_{32} + \Delta D_{21}.$$



Graphs representing the following regions of configuration space from the relationships in figure 3.6.4:

(i) $C_{21}(A)$ (iii) $C_{21}(B)$
(ii) $C_{32}(B)$ (iv) $C_{31}(B)$

Only two degrees of freedom have been shown. The construction of graph (iv) from graphs (ii) and (iii) has been shown and for clarity has not been hashed.

Figure 3.6.5

For all possible positions of part 3 relative to part 1 there must be a position for part 2 which satisfies the constraints relative to part 1 and relative to part 3. In other words, the subset of configuration space, $C_{31}(B)$, satisfies the following property:

For all positions, $\mathbf{p}_{31}$, in $C_{31}(B)$ there exists $\mathbf{p}_{21}$ in $C_{21}(B)$ and $\mathbf{p}_{32}$ in $C_{32}(B)$ with $\mathbf{p}_{31}$ equal to the vector sum of $\mathbf{p}_{32}$ and $\mathbf{p}_{21}$.

Therefore,

$$C_{31}(B) = \{\mathbf{p}+\mathbf{q}:\ \mathbf{p} \in C_{21}(B)\ ,\ \mathbf{q} \in C_{32}(B)\}.$$

Figure 3.6.5(iv) illustrates the resultant subset of configuration space found by summing those illustrated in (ii) and (iii).

An analytical method which allows the inequalities which describe $C_{31}(B)$ to be derived from those which describe $C_{32}(B)$ and $C_{21}(B)$ will now be described.

In the following, pq stands for either 32 or 21.

Suppose that each inequality in C_{pq} involving ΔD_{pq} is solved for ΔD_{pq}. The resulting inequalities can be divided into two categories: those which provide an upper bound expression for ΔD_{pq} and those which provide a lower bound. A single expression bounding ΔD_{pq} above can be constructed by taking the minimum of these upper bound expressions. It will be denoted by UPPER(ΔD_{pq}). Similarly, a lower bound for ΔD_{pq} can be constructed by taking the maximum of the lower bound expressions and will be denoted by LOWER(ΔD_{pq}).

UPPER(ΔD_{pq}) and LOWER(ΔD_{pq}) can be expressed in terms of SUP and INF as follows, where V is the set of all DOF-variables excluding ΔD_{pq}:

$$\text{UPPER}(\Delta D_{pq}) = \text{SUP}(\ \Delta D_{pq},\ C_{pq},\ V\ ),$$
$$\text{LOWER}(\Delta D_{pq}) = \text{INF}(\ \Delta D_{pq},\ C_{pq},\ V\ ).$$

It will be assumed that these can be written as the minimum or maximum of linear expressions of DOF-variables as follows with summations indexed by i in the set $\{x,y,z,\theta,\phi,\psi\}-\{D\}$:

$$UPPER(\Delta D_{32}) = \min_{j}\left(\sum_{i} \alpha_{ij}\Delta i_{32} \right),$$

$$UPPER(\Delta D_{21}) = \min_{k}\left(\sum_{i} \beta_{ik}\Delta i_{21} \right),$$

$$LOWER(\Delta D_{32}) = \max_{j'}\left(\sum_{i} \alpha'_{ij'}\Delta i_{32} \right),$$

$$LOWER(\Delta D_{21}) = \max_{k'}\left(\sum_{i} \beta'_{ik'}\Delta i_{21} \right).$$

Here, α_{ij}, $\alpha'_{ij'}$, β_{ik} and $\beta'_{ik'}$ are all constants. Each minimum and maximum is indexed by a variable j, j', k or k'. The range of these depends on the number of linear expressions involved in each particular case.

Define SUPVAL and INFVAL to be procedures which return values, as opposed to expressions, bounding a DOF-variable. They are equivalent to SUP and INF with third argument equal to the empty set:

$$SUPVAL(\Delta D_{pq}) = SUP(\ \Delta D_{pq},\ C_{pq},\ \{\}\ ),$$

$$INFVAL(\Delta D_{pq}) = INF(\ \Delta D_{pq},\ C_{pq},\ \{\}\ ).$$

We are now in a position to find a set of inequalities which represent $C_{31}(B)$. This is done by finding expressions that bound each of the DOF-variables, ΔD_{31}, above and below. Hence, $C_{31}(B)$ consists of inequalities with the form

$$\Delta D_{31} \leq \text{expression} \quad \text{or} \quad \text{expression} \leq \Delta D_{31}.$$

All variables occurring in the expressions must represent components of the position of part 3 relative to part 1: they must be subscripted 31.

Summations ($\sum$) are again indexed by i ranging over the set $\{x,y,z,\theta,\phi,\psi\}-\{D\}$. Minimums and maximums are indexed by j, j', k or k' and these range over the same sets that they ranged over in the expansions of UPPER and LOWER. First upper bounds for ΔD_{31} are found. There are two ways that this can be done.

$$\Delta D_{31} = \Delta D_{32} + \Delta D_{21}$$

$$\leq \text{UPPER}(\Delta D_{32}) + \text{UPPER}(\Delta D_{21})$$

$$= \min_j\Big(\sum_i \alpha_{ij}\Delta i_{32} \Big) + \min_k\Big(\sum_i \beta_{ik}\Delta i_{21} \Big)$$

$$= \min_j\Big(\sum_i \alpha_{ij}(\Delta i_{32}+\Delta i_{21}) - \sum_i \alpha_{ij}\Delta i_{21} \Big) + \min_k\Big(\sum_i \beta_{ik}\Delta i_{21} \Big)$$

$$= \min_j\Big(\sum_i \alpha_{ij}\Delta i_{31} - \sum_i \alpha_{ij}\Delta i_{21} \Big) + \min_k\Big(\sum_i \beta_{ik}\Delta i_{21} \Big)$$

$$\leq \min_j\Big(\sum_i \alpha_{ij}\Delta i_{31} - \sum_i \alpha_{ij}\text{INFVAL}(\Delta i_{21}) \Big) + \min_k\Big(\sum_i \beta_{ik}\text{SUPVAL}(\Delta i_{21}) \Big)$$

Note that α_{ij}, β_{ik}, INFVAL(Δi_{21}) and SUPVAL(Δi_{21}) can all be replaced by numbers and so the only variables in this expression have subscript 31. Hence this expression has the required form and is an upper bound for ΔD_{31}. In the last step Δi_{21} was replaced by its SUPVAL and INFVAL. Another upper bound for ΔD_{31} can be found by replacing Δi_{32} by its SUPVAL and INFVAL:

$$\Delta D_{31} = \Delta D_{32} + \Delta D_{21}$$

$$\leq \text{UPPER}(\Delta D_{32}) + \text{UPPER}(\Delta D_{21})$$

$$= \min_j\Big(\sum_i \alpha_{ij}\Delta i_{32} \Big) + \min_k\Big(\sum_i \beta_{ik}\Delta i_{21} \Big)$$

$$= \min_j\Big(\sum_i \alpha_{ij}\Delta i_{32} \Big) + \min_k\Big(\sum_i \beta_{ik}(\Delta i_{21}+\Delta i_{32}) - \sum_i \beta_{ik}\Delta i_{32} \Big)$$

$$= \min_j\Big(\sum_i \alpha_{ij}\Delta i_{32} \Big) + \min_k\Big(\sum_i \beta_{ik}\Delta i_{31} - \sum_i \beta_{ik}\Delta i_{32} \Big)$$

$$\leq \min_j\left(\sum_i \alpha_{ij} \mathrm{SUPVAL}(\Delta i_{32}) \right) + \min_k\left(\sum_i \beta_{ik} \Delta i_{31} - \sum_i \beta_{ik} \mathrm{INFVAL}(\Delta i_{32}) \right)$$

In a similar way it possible to find two lower bounds for ΔD_{31}:

$$\Delta D_{31} \geq \max_{j'}\left(\sum_i \alpha'_{ij'} \Delta i_{31} - \sum_i \alpha'_{ij'} \mathrm{SUPVAL}(\Delta i_{21}) \right) + \max_{k'}\left(\sum_i \beta'_{ik'} \mathrm{INFVAL}(\Delta i_{21}) \right),$$

$$\Delta D_{31} \geq \max_{j'}\left(\sum_i \alpha'_{ij'} \mathrm{INFVAL}(\Delta i_{32}) \right) + \max_{k'}\left(\sum_i \beta'_{ik'} \Delta i_{31} - \sum_i \beta'_{ik'} \mathrm{SUPVAL}(\Delta i_{21}) \right).$$

Therefore, in general two inequalities bounding ΔD_{31} above and two bounding it below are obtained. Since there are 6 DOF-variables, ΔD_{31}, it follows that $C_{31}(B)$ is in general expressed by 24 inequalities with the above forms.

The inference which has been described in this section can be applied iteratively to kinematic chains with more than three links. Suppose, nodes A, B, C and D are linked in a chain by three relationships. The first application of the inference can be applied to the relationships between A and B and between B and C to obtain constraints between A and C. The second application can be applied to the relationships between A and C and between C and D to obtain a relationship between A and D.

3.6.3. Application of the Inferences

This section presents an algorithm that applies the two inferences in an appropriate order to find the total constraints between any two nodes of a network. The basic idea is that each path between the two nodes imposes constraints on the nodes. The effect of each path is found using the summation inference and the combined effect of different paths can be found using the intersection inference.

One problem is that there may be a very large number of paths between the nodes. For example if there are four arcs between nodes A and B and four arcs between nodes B and C then the total number of paths from A to C is sixteen. Any reduction in the number of paths is welcome since in a complicated network the paths may be very numerous. One method is to replace sets of arcs between the same two nodes by a single arc. The application of the intersection inference will do this. If this is done for every pair of parts between which there are multiple arcs then the number of paths between the nodes is greatly reduced.

Suppose that the nodes between which constraints are to be derived are denoted A and B. The algorithm has the following stages.

1. Replace sets of arcs between pairs of parts by a single arc using the intersection inference.
2. Find all paths from A to B.
3. For each path apply the summation inference as many times as is necessary to find the constraints implied by that path on A and B.
4. Lastly, apply the intersection inference to the constraints derived from all paths.

There could be a large number of paths despite the first step of the algorithm. This will be true especially in the networks that appear in subsequent chapters. It is possible to reduce the size of the search, however, by a "branch and bound" technique. Paths can be ignored if they provide constraints which are obviously weaker than previously considered paths. For example, a path would provide no constraints if it contained two contacts between planar features with one contact perpendicular to the other. When a path does provide constraints a rough estimate of numeric bounds could be made of the bounds on all variables. These could then be compared with a rough estimate of the bounds implied by previous paths.

The search should be made depth first and the constraints evaluated for each path in turn. When a path has been processed its constraints can be intersected with the constraints from other paths already processed. However, if, during processing of a path, the constraints are found to be weaker than what is already known then the rest of the path can be ignored. Other paths which share the relationships examined in the discarded path can also be ignored.

3.7. IMPLEMENTATION

A system has been implemented based on the work in this chapter. The input to the system is a description of a set of parts. Each part is represented as a set of features with given position relative to some central coordinate system of the part. Each feature has an extent. A list of pairs of features which can come into contact is input. The nominal position of each part is given.

From this information the system works out bounds on the positions of the parts. The following are some of the calculations which it performs.

- Find the extent of overlap of two features given the nominal positions of their respective parts.
- Find the constraints implied by the contact between two features. There is a different set of constraints for each type of feature. Initially, the constraints are expressed with respect to the feature coordinate systems.
- Make change of variables in the constraints so that they are expressed with respect to the main coordinate systems of the respective parts.
- Given two sets of constraints between parts A and B find

a set of constraints which expresses the total effect of the original constraint sets. This is trivial and merely involves taking the union of the constraint sets.

- Given constraints between parts A and B and between parts B and C find the resulting constraints between parts A and C. This assumes that a common coordinate system in part B is used for the two sets of constraints.
- Given a set of constraints involving many degree of freedom variables find the upper and lower bounds which it implies on any of the variables. This uses the SUPINF algorithm (appendix 2).

So far, the implementation has been performed for two-dimensional geometry only. No path finding has yet been included. A depth first search would be preferable to a breadth first search. This would allow paths to be rejected if it is found that after a partial examination they provide very weak or no constraints.

3.8. CONCLUSION

This chapter started by describing the object-level robot programming system, RAPT, which provides a basis for some of the work described in this thesis. The representation of nominal geometry in an uncertainty reasoning system could closely follow the representation used by RAPT.

It was explained how position uncertainties of perfectly formed parts in an assembly can be found.

The position uncertainty of each part is represented by a set of inequalities involving the degree of freedom variables of the parts. The form of the inequalities depends on what coordinate system is chosen. It is useful to be able to convert a set of inequalities representing an uncertainty in one coordinate system to a set of inequalities representing the same uncertainty with respect to a different coordinate system.

Each potential contact between features is converted to constraints on the relative positions of the features. The constraints thus derived are combined by two inference techniques. The first takes constraints from parallel relationships and outputs their combined effect. The second takes a chain of parts with relationships between them and outputs constraints between parts at the ends of the chain. Using these techniques it is possible to find constraints on the position of parts in a structure of arbitrary complexity.

The following similarities exist with the work of Taylor (1976) (see section 2.2).

- A part is represented as a collection of features.
- Relationships between pairs of features represent possible contacts.
- Relationships give rise to constraints in the form of inequalities.
- The constraints are analysed to determine possible variations in the position of the parts.

The main differences in this thesis from Taylor's work are the following.

- Taylor does not analyse imperfect contacts. He derives constraints from the overlap between contacting features or inputs the constraints into the system directly.
- He does not formalise the analysis of assemblies of arbitrary complexity.
- He uses linear programming techniques for analysing linear constraints and specialised inferences for analysing non-linear constraints. The SUPINF algorithm

has the advantage that it can be generalised to non-linear constraints.

- The network of parts and potential contacts has no counterpart in Taylor's work.

Although the work presented here has similarities to Taylor's work it is more general in terms of the complexity of the kinematics that can be handled. Contacts between surfaces are not assumed to be perfect and all constraints are derived from geometry. The techniques used for analysing the constraints are more general. In chapters 5 and 6 similar techniques will be used to analyse the networks that arise in the presence of toleranced parts.

Chapter 4: INTRODUCTION TO TOLERANCES

This chapter introduces tolerances, the ways they can be allocated and a theory of geometrical tolerances. The first section deals with traditional methods of tolerancing and the standards used. The second section explains a formalism of tolerances developed by Requicha at the University of Rochester (1983a). This formalism is used in chapters 5 and 6 to build a computational representation of tolerances. The following section discusses a selection of related topics.

The ultimate goal of providing tolerance information is to state what variations in the shape of a part are acceptable. A major requirement is that the part should function correctly. The tolerances allocated indicate the possible variations that can be tolerated if the part is to function correctly. The tolerances must be large enough to allow the part to be manufactured economically. If there is too much tolerance then there is no guarantee that the part will function properly. If there is too little tolerance then it will be difficult to manufacture the part.

4.1. STANDARD TOLERANCING TECHNIQUES

The standards which engineers use to specify tolerances are defined by British Standards in (BS 308). In this section an outline will be given of the different techniques used.

An engineering drawing contains dimensions to indicate the sizes of features and relationships between features. A dimension can be a distance or an angle. Any dimension may be given a tolerance which indicates an upper and a lower bound for the dimension. Examples of dimensions are the following.

- The distance between two features such as holes;
- The size of a feature such as the length of a shaft or the diameter of a hole.

Dimensions are shown on engineering drawings by double ended arrows spanning the distance which the dimension refers to. The nominal length and tolerance of a dimension are included next to the arrow as 10±0·01 for example. This method of tolerancing will be referred to as plus/minus tolerancing.

A problem with toleranced dimensions is that dimensions are not easily defined on an actual instance of a part. This is because the surface of a manufactured part is not perfectly formed. For example, surfaces which are nominally flat will inevitably contain undulations or roughness. It is impossible to define a unique distance between two such surfaces because the distance varies depending on the precise points between which the measurement is made. The following questions might be asked.

- How can the diameter of a hole be defined when the hole is not perfectly round?
- How can the distance be defined between two holes which are not perfectly round?
- How can the distance between two imperfect planes be defined?

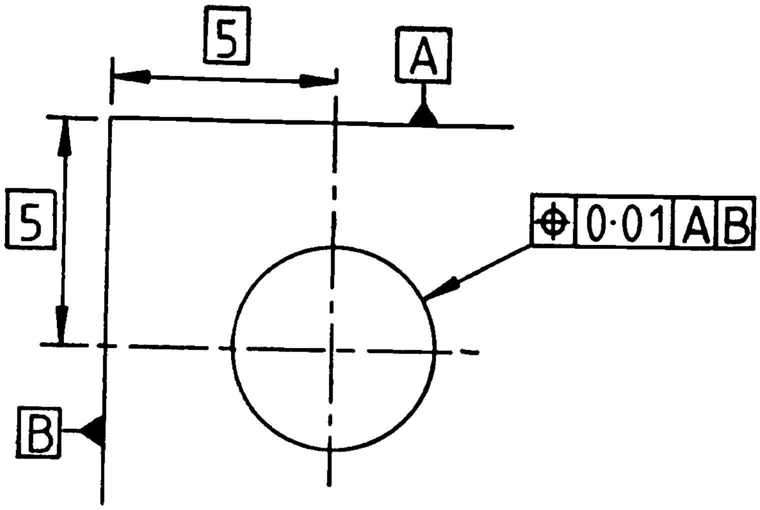
There are standard techniques for defining tolerances which allow these problems to be bypassed. The techniques are called "geometric tolerancing" and are described in British Standards 308, part 3 (BS 308). Use is made of <u>tolerance zones</u>, which are regions of space, in which the surface of a feature must lie. They allow the size and position of a feature to be constrained in a simple way even allowing for the feature's shape to be irregular. For example the size of a feature can be defined by a zone which has a specified size and a specified thickness. The real surface has to lie inside this zone.

The well-formedness of a surface can also be specified by a tolerance zone. For example, tolerance of flatness applied to a nominally flat surface states that the actual surface must lie between two planes separated by a given distance. A tolerance of cylindricity applied to a nominally cylindrical surface states that the actual surface must lie in a cylindrical zone with a given thickness.

The position tolerance given to a feature is often defined at the maximum material condition (MMC) of the feature. This implies that the stated position tolerance holds only when the feature contains maximum material. In the case of a cylindrical shaft the maximum material condition occurs when the shaft has maximum diameter. In the case of a cylindrical hole the maximum material condition occurs when the hole has minimum diameter. If the feature has a diameter which differs from the diameter corresponding to maximum material by a certain amount then its position tolerance can be increased by an equal amount. For example consider a shaft with diameter 1·0±0·01 and a position tolerance of 0·02 at MMC. MMC occurs when the diameter is 1·01. If the shaft actually has a diameter of 0·995 (ie. 0·015 less than MMC) then the allowable variation in position is 0·02 + 0·015 = 0·035.

The position of a feature may be defined relative to other features by using datums. A datum is an imaginary surface or line fixed in a part and located relative to some surface feature. Three examples of datums are (1) the axis of a hole, (2) the plane defined by a planar feature and (3) the plane of symmetry of two parallel holes.

The position of a feature is usually defined relative to a set of datums. For example, figure 4.1.1 shows how a hole with a position tolerance of 0·01 relative to two datum faces A and B appears on an engineering drawing. The tolerance is indicated by a box with a symbol to indicate the type of tolerance, a value to indicate the amount of tolerance and letters to indicate the relevant datums.



An example to show how tolerances are indicated on an engineering drawing. The hole has a position tolerance relative to datums A and B.

Figure 4.1.1

The use of datums in the design specification may reflect the process of manufacture. While a feature is being cut the part must be supported somehow. Imperfections in the surfaces supporting the part will affect the position of the feature being cut. Hence, the feature ends up being positioned relative to the supporting features. The use of datums can also reflect the function of a part. It may be important that two holes are positioned accurately relative to one another so that they can be used to attach another part. Their position relative to some other feature may not be so important.

4.2. TOLERANCE SEMANTICS

Requicha (1983a) has formalised and generalised the standard practices described above. Such a formalism is necessary before an attempt can be made to represent tolerances computationally and before inferences about toleranced parts can be made. Therefore, a detailed account of Requicha's paper is given here. It is important to understand the formalism before proceeding with the rest of this thesis.

4.2.1. Variational Classes

A variational class of a part is a set of objects which are acceptable instances of the part. They differ only in ways allowed by the tolerance specification and so are all functionally equivalent. A variational class of a part is specified by its tolerance specification. The semantics of tolerances are described by mathematical procedures which are used to check the membership of an object in the variational class.

4.2.2. Features

The surface of a nominal part is divided into two-dimensional regions called nominal features. The surface of an instance of the part is divided into real features each one corresponding to a nominal feature. For each nominal feature there is a set of assertions that the corresponding real feature must satisfy. These assertions contain the tolerance information.

It is often necessary to attach a set of assertions to a group of features. In this case a set of features are taken together as a composite feature. The assertions are applied to the composite feature. The term "simple feature" is used to refer to a feature which is not composite.

Each feature has a corresponding extended feature. This is a possibly infinite solid whose surface contains the nominal feature. The extended feature must have material on the same side that the nominal feature has material. Its surface should be as simple as possible whilst satisfying the property that it contains the nominal feature.

Examples of nominal features and their corresponding extended features are shown in figure 4.2.1. The extended feature associated with a cylindrical shaft is an infinitely long cylinder whose surface coincides with the nominal feature. A feature consisting of two plane faces joined at an angle has an extended feature which is the intersection of two planar half-spaces whose surfaces are coplanar with the nominal surfaces and which have material on the same side that the nominal faces have material.

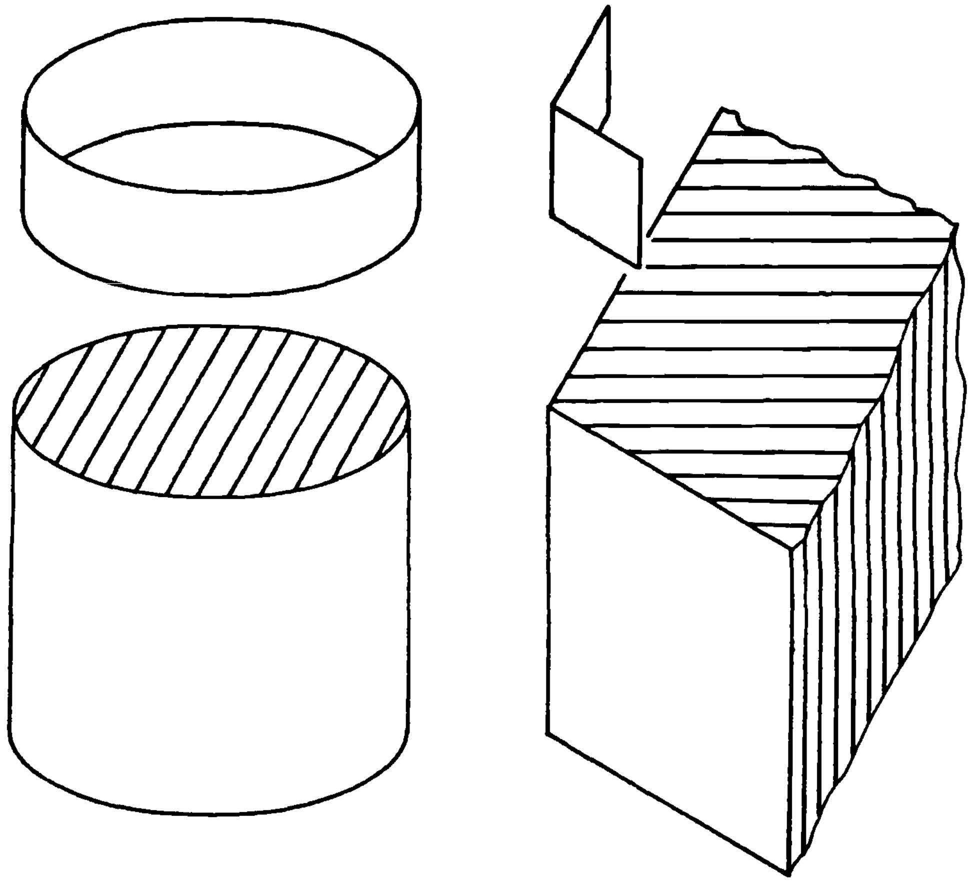
A more precise definition of an extended feature can be given in terms of constructive solid geometry (CSG)*. The extended feature is built from unions and intersection of primitive half-spaces. No half-space may be included if it contributes to the surface of the extended feature but not to the surface of the nominal feature. The material side of a half-space must be the same as the material side of the portion of nominal feature to which it contributes.

Not all surfaces can define extended features in this way. Such surfaces are not valid features. An example is a pair of parallel, but not coplanar, planes with their normals pointing in the same direction.

4.2.3. Datums

A datum is an infinite plane, an infinite straight line or a point embedded in a part. It may be defined by any nominally

* Constructive solid geometry see Requicha (1983a).



Examples of two nominal features (above) and the corresponding extended features (below).

Figure 4.2.1

symmetric feature. The datum is the plane, line or point of symmetry of the feature. A datum may also be defined by a planar feature and in this case the datum is the infinite plane coplanar with the feature.

There is a measuring procedure used to define a datum on a real feature when the feature is symmetric. A copy is taken of the associated extended feature and is scaled and oriented so that it contains the real feature. The scaling factor is chosen to be the smallest possible which can create this condition. The orientation is chosen such that the scaling factor can be as small as possible. The plane, line or point of symmetry of the scaled copy is the datum for the real feature. Sometimes, centrepoints, lines or planes defined in this way are called measured entities.

In chapter 5 the geometry of the measuring procedure will be analysed. The scaled and oriented copy of the extended feature will then be referred to as the measuring solid.

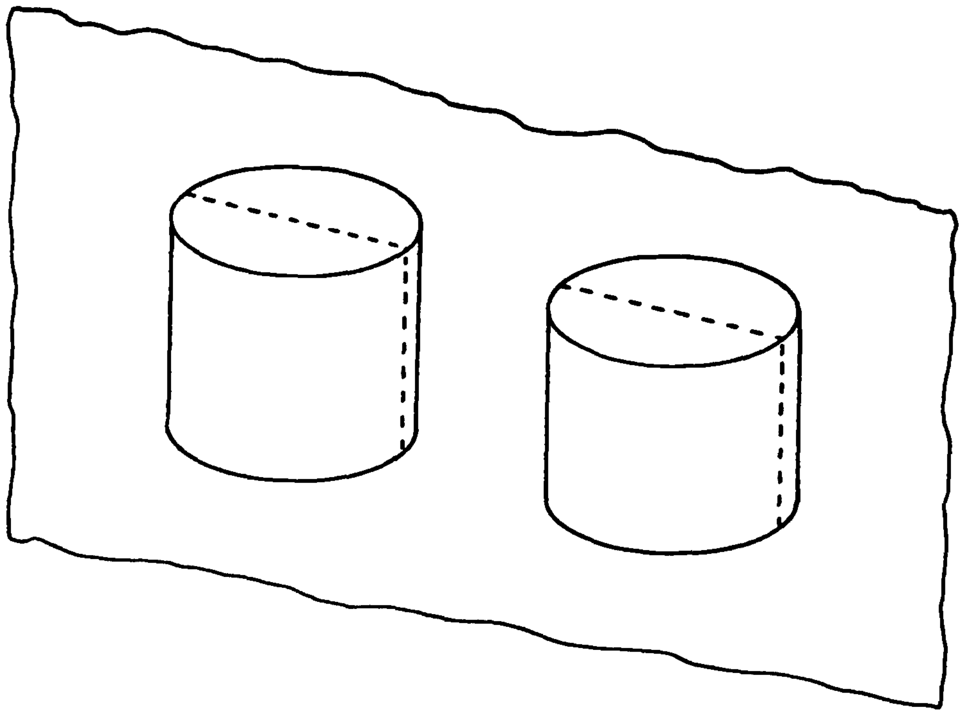
For a planar feature the datum is taken as a plane which just rests against the real feature. Its position is chosen so that the volume between the datum and the real feature is reduced to a minimum.

These procedures do not define a unique datum for all real features. There may be several suitable positions of the measuring solid with the same minimum size. However, the measuring procedure does allow bounds on the possible positions of a datum to be determined for a feature with a given tolerance specification. Such bounds will be determined in chapter 5.

A datum can be defined by any composite feature as long as it is symmetric. For example, two parallel cylindrical pegs could be treated as a composite feature and could define a datum as shown in figure 4.2.2.

A group of datums is sometimes treated as a single unit referred to as a system of datums.

Sometimes a system of datums is ordered. The datums are defined sequentially starting with the primary datum. It is located using the measuring procedure on its associated feature, or in the case of a planar feature, by resting the plane against the feature, as already described. Subsequent datums are located relative to their associated features in the same way but with the additional constraint that their position relative to datums already defined must be "correct". That is, the relative position of all datums in the system must be the same as the relative position of their corresponding nominal features.



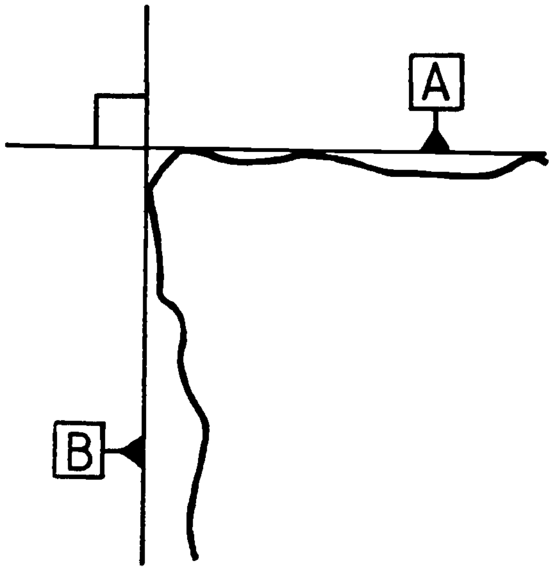
A composite feature consisting of two cylinders and a planar datum defined by it.

Figure 4.2.2

Figure 4.2.3 shows an example of an ordered system of two datums. The primary datum rests against its associated feature. The secondary datum also rests against its associated feature but with the additional constraint that it is perpendicular to the primary datum. It only manages to contact its associated real feature at one point.

4.2.4. **Offset Solids**

The definition of tolerances uses "offset solids". These are derived from a solid by growing it or shrinking it by a small amount. A positive offset means that the solid has been grown and a negative offset that it has been shrunk. For a non-negative offset, d, the offset solid of solid, S, may be written formally as



A datum-system in which A is primary and B is secondary.

Figure 4.2.3

$$O(d;S) = \{p: dist(p,S) \le d\} \quad \text{where } d \ge 0.$$

The function "dist(p,S)" is defined as the distance from p to the point of S which is closest to p.

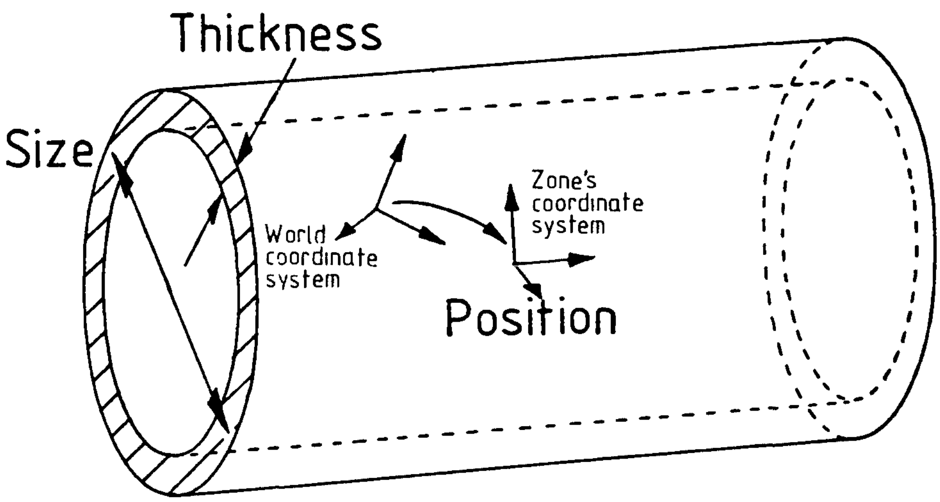
For negative offsets the complement of S is grown by a positive amount and then the complement taken of the result giving

$$O(d;S) = comp\ O(-d; comp\ S) \quad \text{where } d<0.$$

"comp" is used here to represent taking the complement of a set.

4.2.5. Tolerance Definitions

The definitions of tolerance types described here are applicable to any shape of feature and are therefore more general than those of standard tolerancing practice. The general principle for tolerance definition is that the real feature must lie inside a "tolerance zone". Most tolerance zones are produced by taking the set difference of two offset solids of an extended feature. The result is a shell of finite thickness similarly shaped to the surface of the extended feature. For example the tolerance zone of a cylindrical feature is an infinite cylindrical shell (figure 4.2.4). The thickness, diameter, orientation and position of the zone may be defined or undefined depending on the type of tolerance.



A tolerance zone.

Figure 4.2.4

Different tolerance types will be defined below using the following notation.

F is a nominal feature,

G is the real feature corresponding to F on a certain instance of the part,

H is the extended feature corresponding to F.

The symbol "$\subseteq$" is used to represent set inclusion.

Form Tolerance

A form tolerance specifies how much roughness of the surface is permitted. The real feature must lie in a tolerance zone with a specified thickness. The position of the tolerance zone is undefined. A form tolerance with parameter T_f has a zone constructed from two offset solids with offsets whose difference is T_f. Formally this is expressed by

$$G \subseteq O(d_1;H')-O(d_2;H') \quad \text{where} \quad d_1-d_2 = T_f,$$

where H′ is a congruent instance of H. Hence H′ is the result of H subjected to some unspecified rigid transformation. This ensures that the position of the zone is undefined.

Size Tolerance

A size tolerance is defined using two parameters, S_1 and S_2, $(S_1>S_2)$ used as offsets of two offset solids. The position of the tolerance zone is undefined. The real feature must lie between the surfaces of the offset solids. Formally,

$$G \subseteq O(S_1;H')-O(S_2;H').$$

Again H′ is a congruent instance of H. Note that size tolerance is equivalent to form tolerance when applied to a planar feature.

Position Tolerances

One type of position tolerance is the absolute position tolerance. This is a very strict form of tolerance. The real feature must lie in a tolerance zone whose size, thickness and position are all defined. The position is fixed relative to a specified system of datums. Formally, an absolute position tolerance with parameter P is defined as

$$G \subseteq O(P/2;H') - O(-P/2;H'),$$

where H′ is congruent to H and is **correctly positioned** with respect to a certain specified system of datums. This means that the position of H′ relative to the datums must be the same as the position of F relative to the nominal features corresponding to the datums.

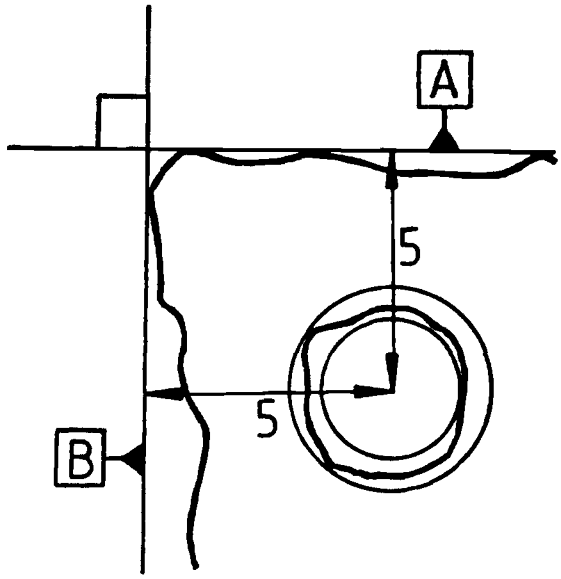
A maximum material condition (MMC) position tolerance (described in section 4.1) is defined as follows. Suppose the feature has a size tolerance with parameters S_1 and S_2. Then a MMC position tolerance with parameter P implies that

$$G \subseteq O(S_1+P;H'),$$

where H′ is congruent to H and correctly positioned relative to a specified system of datums. The idea behind standard MMC-position tolerances, that the variation in position of the feature increases as the amount of material in the feature decreases, holds in this definition.

An example of the way datums are used is shown in figure 4.2.5. This is an exaggerated illustration of the real surfaces, datums and tolerance zone implied by the drawing in figure 4.1.1. The hole lies nominally at a distance of 5 units from two perpendicular faces. The zone of position tolerance must therefore be 5 units from each of the datums associated with the two faces.

A feature can define a datum which in turn can define the position of other features. The features and datums and the relationships between them form a network: the nodes represent features and datums and the arcs represent relationships. In chapter 5 the network will be discussed in more detail and it will



An exaggerated real part showing datums and tolerance zone. The zone lies at a location fixed relative to the datums.

Figure 4.2.5

be shown how the network can be analysed to find how tolerances on one feature affect another feature.

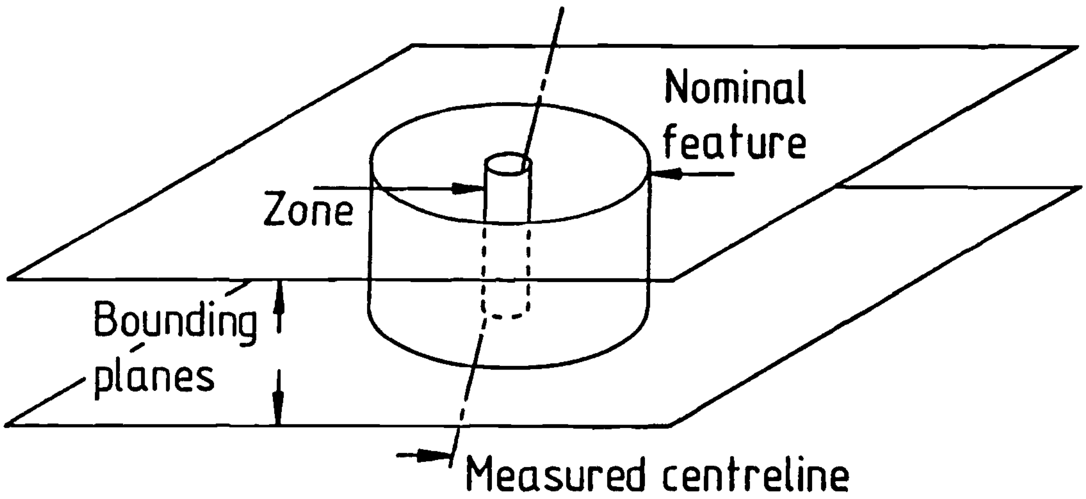
Orientation Tolerance

An orientation tolerance constrains the orientation of a feature but not its position. The tolerance zone has fixed thickness and orientation but undefined position and size. If the tolerance has parameter, Q, then

$$G \subseteq O(d_1;H')-O(d_2;H') \quad \text{where} \quad d_1-d_2 = Q$$

and H' is correctly orientated but not necessarily correctly positioned relative to a specified system of datums.

The next two types of tolerance have not been dealt with in the rest of the thesis but have been included here for completeness. The reader could, therefore, skip these if desired and go straight to section 4.3.



The definition of a "regardless of feature size" (RFS) position tolerance.

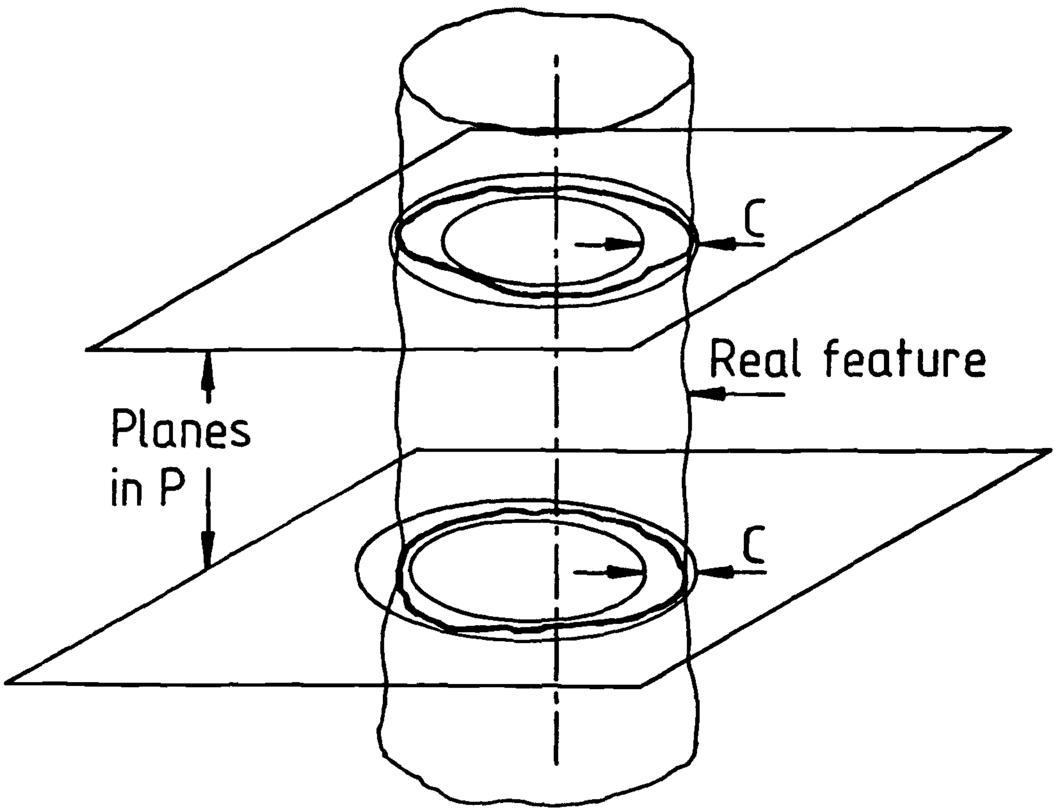
Figure 4.2.6

Regardless of Feature Size (RFS) Position Tolerance

An RFS-position tolerance constrains the position of a feature without constraining its size. The tolerance is interpreted in Requicha's formalism as follows.

Basically the tolerance is defined by putting constraints on the position of a measured entity defined by the feature. (Recall that a measured entity is equivalent to a datum and is defined by the measuring procedure.) The position of the measured entity is constrained by a zone which has finite extent unlike tolerance zones described previously. The zone is correctly positioned relative to some specified system of datums. It has an extent which is bounded in the same way as the extent of the nominal feature.

For example, consider a cylindrical feature. It defines a measured centreline. The nominal cylinder is bounded by two planes as shown in figure 4.2.6. If the tolerance parameter is P then the zone is a cylinder with diameter P and bounded by the planes that bound the nominal feature. It has correct position relative to a specified datum-system. The portion of the measured centreline of the real feature that lies between the bounding planes must lie inside the zone.



The definition of a curve tolerance.

Figure 4.2.7

Curve Tolerances

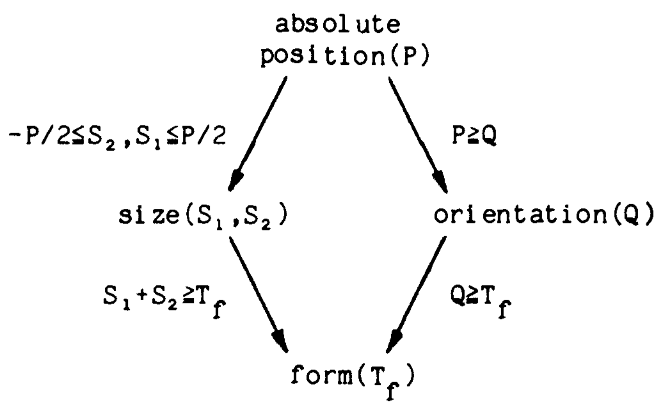
A curve tolerance applied to a cylinder is equivalent to the tolerance of roundness used in standard practice. Two-dimensional tolerance zones are used (figure 4.2.7). Let P be the set of all planes perpendicular to some axis of the feature. This will usually be a measured entity defined by the feature. A real feature satisfies a curve tolerance with parameter C if the result of intersecting the real feature with any plane in P lies in a two-dimensional zone which is the intersection of P and a three-dimensional zone defined as for form tolerances but with a parameter, C. A zone with different diameter and position can be used for each plane in P. Planes which do not intersect the real feature can be ignored.

4.3. CONSTRAINTS BETWEEN TOLERANCE TYPES

The following tolerance types have all been defined with tolerance zones which are the set difference of two offset solids: form, size, orientation and absolute position. Whenever more than one of these are applied to the same feature there are constraints which their parameters can be assumed to satisfy.

Some types of tolerance imply others. For example a feature with a size tolerance with parameters S_1 and S_2 implies that the feature has a form tolerance with parameter S_1-S_2. Therefore, there is no point in applying an explicit form tolerance unless it has a parameter less than S_1-S_2. However, to simplify the computational representation of a tolerance specification (the subject of chapter 5) it is convenient to assume that all features have a form tolerance. If a feature has size tolerance but no explicit form tolerance then it can be understood that it has form tolerance with parameter S_1-S_2. In a similar way an orientation tolerance implies a form tolerance and an absolute position tolerance implies form, size and orientation tolerances. The implications between tolerance types are shown in figure 4.3.1. Next to the arcs are shown the constraints that their parameters can be assumed to satisfy. T_f, Q and P are the parameters for form, orientation and position tolerances respectively and S_1 and S_2 are the parameters for size tolerance.

Since all features must have their position constrained it follows that all must have a position tolerance. As a result, it can be assumed that the four tolerance types, form, size, orientation and absolute position, are present on all features.

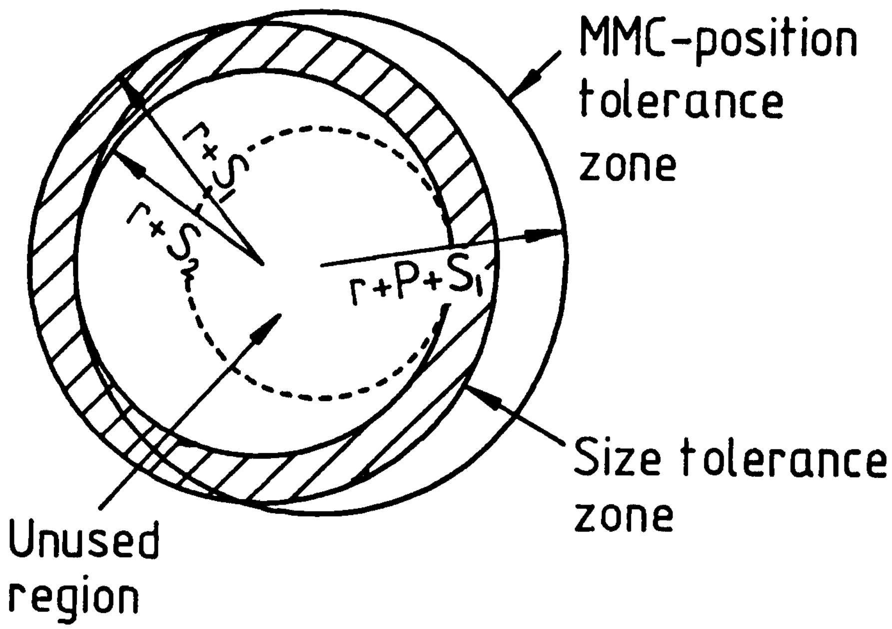


Implications between tolerance types. Parameters for each tolerance type are shown in brackets. Constraints are shown beside the arcs.

Figure 4.3.1

MMC-Position Tolerances

The above diagram does not include MMC-position tolerance though this is an important tolerance type. Rather than include the absolute and MMC position tolerances as separate types of tolerance it is simpler to generalise them to a single **position tolerance**. The MMC-position tolerance is defined using only one offset-solid whilst the absolute position tolerance is defined as the set difference of two offset solids. However, a MMC-position tolerance zone has a region which cannot be entered by the surface of the real feature since the feature must also satisfy a size tolerance. This is illustrated for a cylindrical feature in figure 4.3.2. This region can be removed without affecting the class of possible real features allowed by the zone. The result is a zone with similar topology and properties to the zone of absolute position tolerance.



A zone of MMC-position tolerance and a zone of size tolerance (shown hashed) associated with the same feature. The size tolerance zone is at its maximum displacement, $P+S_1-S_2$, relative to the other zone. There is a region of the MMC-position tolerance zone which cannot be entered by a real feature.

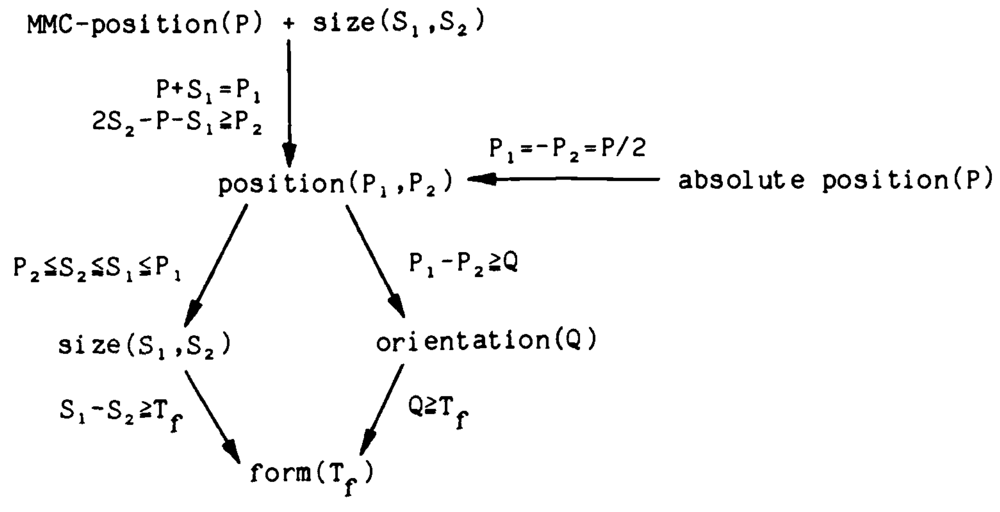
Figure 4.3.2

It follows that a generalisation of both the MMC and the absolute position tolerances can be defined by a zone,

$$O(P_1;H') - O(P_2;H').$$

Here H' is a copy of the associated extended feature with correct position with respect to specified datums. The MMC-position tolerance requires P_2 to be chosen small enough so that the region which is removed does not affect the class of possible real features. In figure 4.3.2 it can be seen that for a cylindrical feature P_2 should be less than or equal to $2S_2-S_1-P$. P_1 should be $P+S_1$. An absolute position tolerance with parameter P is obtained by letting $P_1=-P_2=P/2$.

Including the MMC-position tolerance in figure 4.3.1 we obtain figure 4.3.3.



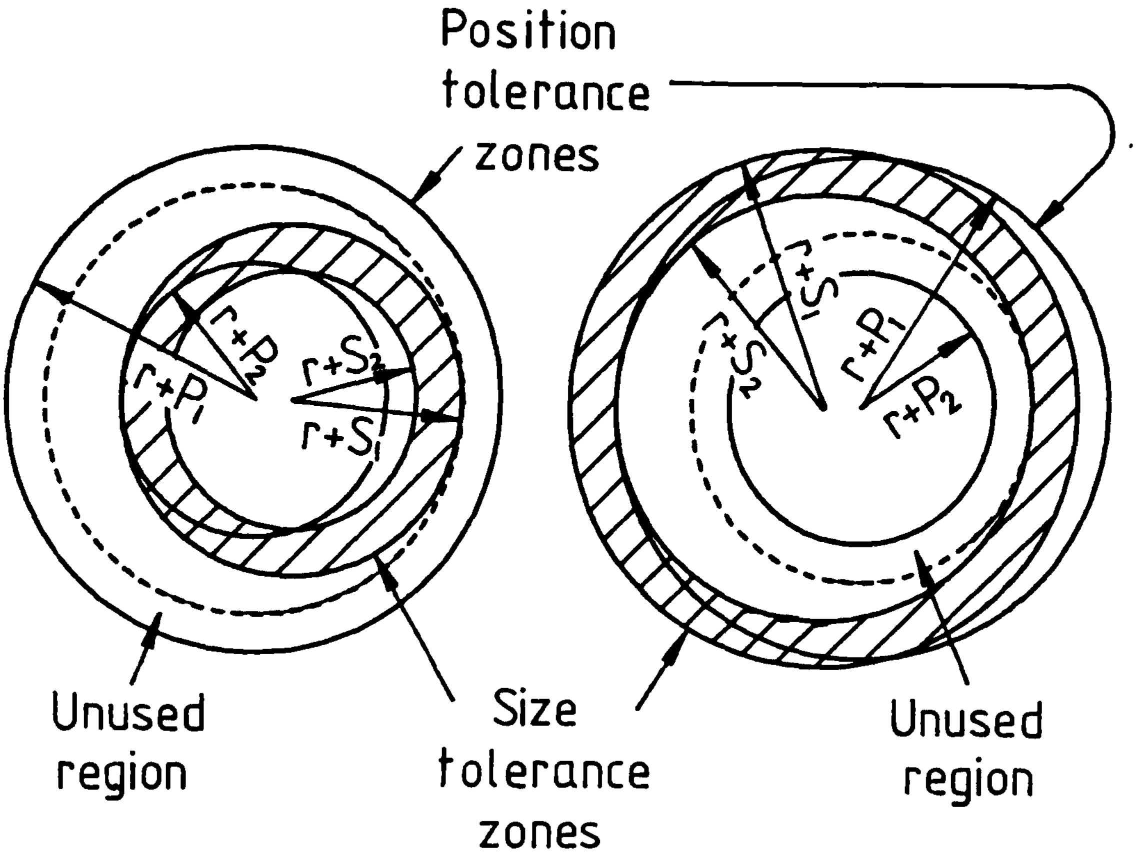
Implications between tolerance types when MMC-position tolerance is included.

Figure 4.3.3

Further Constraints Between Tolerance Parameters

There are other less obvious constraints between tolerance parameters. Suppose we have tolerances of size and position applied to the same feature with zones

$$O(P_1;H')-O(P_2;H') \quad \text{and} \quad O(S_1;H'')-O(S_2;H'').$$



Constraints between size and position tolerance parameters. The size tolerance zone is shown hashed and has maximum displacement relative to the position tolerance zone.

Figure 4.3.4

Figure 4.3.4 shows two situations (concerning a cylindrical feature) in which a zone of position tolerance has unnecessary thickness. In the first the maximum possible displacement of the size tolerance zone relative to the position tolerance zone is S_1-P_2 and in the second it is P_1-S_2. In both cases there is an unused region of the position tolerance zone that the size tolerance zone cannot intersect. These are bounded by dashed lines in figure 4.3.4. In the first case the position tolerance parameter, P_2, can be replaced by $S_2-(P_1-S_2)$ if this is larger than P_2. In the second case P_1 can be replaced by $S_1+(S_1-P_2)$ if this is less than P_1. Hence it can be assumed that

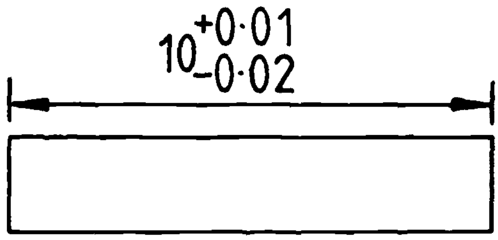
$$2S_2-P_1 \leq P_2 \quad \text{and} \quad P_1 \leq 2S_1-P_2.$$

Similar constraints could be obtained for other types of feature.

4.4. DIMENSIONS BETWEEN TWO FEATURES WITH NO PREFERRED DATUM

Requicha states that all features have their position defined relative to a datum and that in general there is a tree of datums and features. At the top of this tree is a master datum. Therefore, all features are ultimately dependent on the master datum.

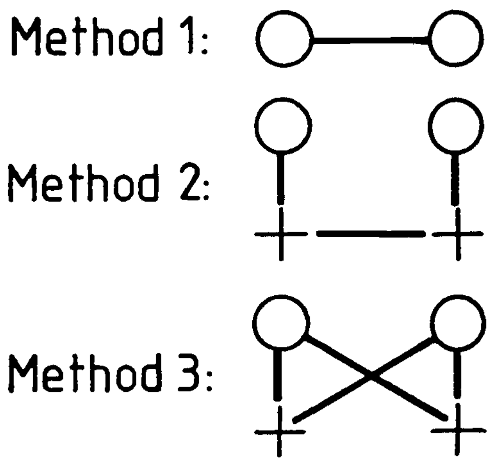
A contradiction to this statement occurs, however, when a part has a dimension between two features as in figure 4.4.1. In this case there is no concept of a master datum. The relationship between the features is symmetrical and it seems unreasonable that one feature should be chosen as a datum for the other.



A tolerance between two faces in which there is no master datum.

Figure 4.4.1

Three methods of interpreting this situation in terms of tolerance zones and measured entities (datums) were considered. Each can be represented as a network of zones and datums in a way which is compatible with the representation to be built up during the next chapter. Firstly, each feature could lie in a zone of position tolerance. The sum of the parameters of each zone should equal the total variation allowed by the dimension. The distance between the zones must be equal to the midpoint of the range of the dimension. The situation can be represented as a network (figure 4.4.2, method 1) by two nodes to represent the zones connected by an arc which contains constraints fixing the distance between the nodes.



Methods of representing a position tolerance between two features with no master datum. The circle nodes represent tolerance zones and the cross nodes represent measured entities.

Figure 4.4.2

For example suppose there is a dimension of $10^{+0\cdot01}_{-0\cdot02}$ between two features. It is equivalent to a dimension of 9•995±0•015. The tolerance would be satisfied if the features were contained in zones with thickness 0•15 separated by 9•95.

Secondly, the dimension could constrain the relative position of measured entities defined by each feature. The distance between the two datums should be within the range of the toleranced dimension. This constraint is satisfactory with point datums. However, in the case of linear and planar datums which may not be parallel the constraint should only be applied to a portion of each corresponding to the extent of the nominal feature. The result can be represented by the network shown in figure 4.4.2, (method 2). Each of the measured entities is shown linked to its associated feature. The measured entities have an arc between them to represent the constraint on their relative position. This method has similarities to the RFS-position tolerance: the sizes of the features are not constrained.

Thirdly, each feature could be positioned relative to a measured entity defined by the other feature. This can be represented by the network shown in figure 4.4.2, method 3. In figure 4.4.1 the right-hand plane would lie in a tolerance zone correctly positioned relative to a datum defined by the left-hand plane. Conversely the left-hand plane would lie in a tolerance zone correctly positioned relative to a datum defined by the right-hand plane. The thickness of each zone should be equal to the total variation allowed by the dimension.

The first method produces the simplest network, only one arc being needed to describe the relationship between the features. The constraints implied by the second method do not constrain the size of the features and so this is similar to a RFS-position tolerance. The third method produces a more complex network and has no obvious connection with standard tolerancing practice.

4.5. DIFFERENT FEATURE ALLOCATIONS AND TOLERANCE SPECIFICATIONS

There are many ways in which the surface of a part can be divided into features. The only limitation on the shape of a feature is that an **extended feature** must be definable from it. The feature allocation is an important component of a tolerance scheme. Different tolerance schemes might be equivalent but it is an open problem to decide whether they are or not.

For instance, consider a slot consisting of two parallel planes. The slot can be considered as a single feature and can be given a size tolerance. Alternatively each plane can be thought of as a separate feature and a position tolerance can be applied between them. In both schemes the distance between the planes and their relative orientation are constrained. It seems that the two tolerancing schemes are equivalent.

Broadly, there are two types of tolerance: those that act between features and those that act within a feature. In the first category are position and orientation tolerances and in the second are form and size tolerances. Position tolerances define the position of a feature relative to a datum which is centred on another feature. Size and form tolerances act on single features. A position tolerance on a feature A relative to a feature B applies constraints similar to a size tolerance on the composite feature consisting of A and B. It seems that there would be schemes with a maximal number of tolerances between features and schemes with maximal number of tolerances within features. The former would be associated with a feature allocation with a maximal number of simple features and the second with a maximal number of composite features. In chapter 5 the methods by which tolerances are allocated to composite features will be considered.

4.6. CONCLUSION

This chapter has briefly described the standard techniques, dimensionsal tolerancing and geometric tolerancing, used by engineers to define tolerance specifications. A formalism and generalisation of geometric tolerances developed by Requicha has been described. This formalism has been used as a foundation for the work described in the next two chapters.

In Requicha's formalism there are basic tolerance types defined in such a way that they are applicable to features of any shape. Most tolerance types use zones which are the set difference of two offset solids. Exceptions to these are MMC-position tolerance, RFS position tolerance and curve tolerances. The latter two have not been dealt with in the rest of this thesis. It has been shown that MMC-position tolerances can be expressed equivalently by zones which are the set difference of two offset solids. Hence, only zones with this form need be considered from now on.

The use of datums has been formalised. A method was explained by which their position can be defined relative to a real feature with imperfect form.

It was mentioned that networks of features and datums arise. A feature can been located by a datum centred on some other feature which is in turn located by another datum and so on. A master datum ultimately defines the positions of all features. However, dimensions between pairs of features create a situation where there is no obvious master datum.

Various constraints that exist between the parameters of different tolerance types applied to the same feature were explained (section 4.3). These will simplify the computational representation of tolerances explained in the next chapter by allowing default values to be given to tolerance types that have not been specified explicitly.

Chapter 5: REPRESENTING TOLERANCED PARTS COMPUTATIONALLY

Introduction

This chapter describes how a toleranced part can be represented in a computer. The basic idea is to construct a network of tolerance zones and datums linked by arcs containing constraints on their relative positions. This will be called a zone-datum network. Much of this chapter will explain the properties of geometric relationships between tolerance zones and datums. Each relationship gives rise to constraints which are attached to arcs in the zone-datum network.

A tolerance specification describes a set of regions of three-dimensional space that could be occupied by valid instances of the part. The set is called a variational class. A representation of a toleranced part stored in a computer attempts to describe the same variational class.

An assumption to be made from now on is that a tolerance specification is only useful if all copies of the part that it defines are functional. Hence any required property of the toleranced part must be true for all copies of the part that satisfy the tolerance specification. This is because statistical distributions of tolerances is an area not covered by this thesis. A measurement or a calculation that applies to only some instances of a part is useless without statistical information. Although, for example, it might be satisfactory if only 99% of parts manufactured are functional and for there to be 1% wastage it is impossible to know, without statistical information, what these percentages are. Hence all measurements and calculations must apply to 100% of parts.

The different situations in which constraints arise will be discussed. Each will be explained in terms of its representation in the zone-datum network. It will be shown how the representation can be analysed to obtain information about the toleranced part. The first section gives an example to show situations where relationships occur.

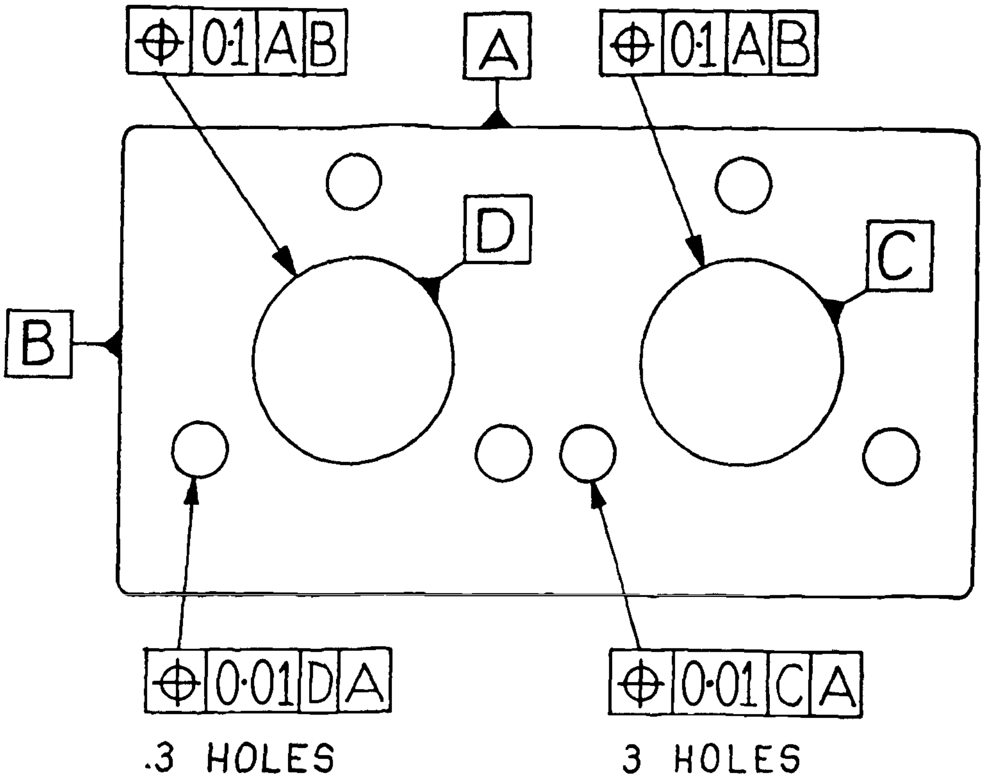
5.1. EXAMPLE

An example is presented here which shows many of the ways that geometric relationships occur. The part shown in figure 5.1.1 is explained and a network of its features and datums is presented as a preliminary to the zone-datum network.

The part is a plate with two groups of four holes. The function of the holes is to attach two dials. Therefore, the holes in each group must be positioned accurately if they are to meet up with holes in the dials. However the relative position of the two dials is not critical and so the relative position of the two hole groups need not be defined so accurately.

An ordered datum system is defined by two sides of the plate. The primary datum A is associated with the horizontal side of the plate and the secondary datum B with the vertical side. They are constrained to be perpendicular since the nominal features are perpendicular.

The two large holes are positioned with respect to datums A and B with a relatively large tolerance of 0·1. Each defines a line datum, denoted by C and D, corresponding to its axis of symmetry. There are three small holes round each large hole with a position tolerance of 0·01. Three of the small holes are positioned relative to the datum-system consisting of primary datum C and secondary datum A. The distance of their position tolerance zones from C is fixed and the angle of the line from C

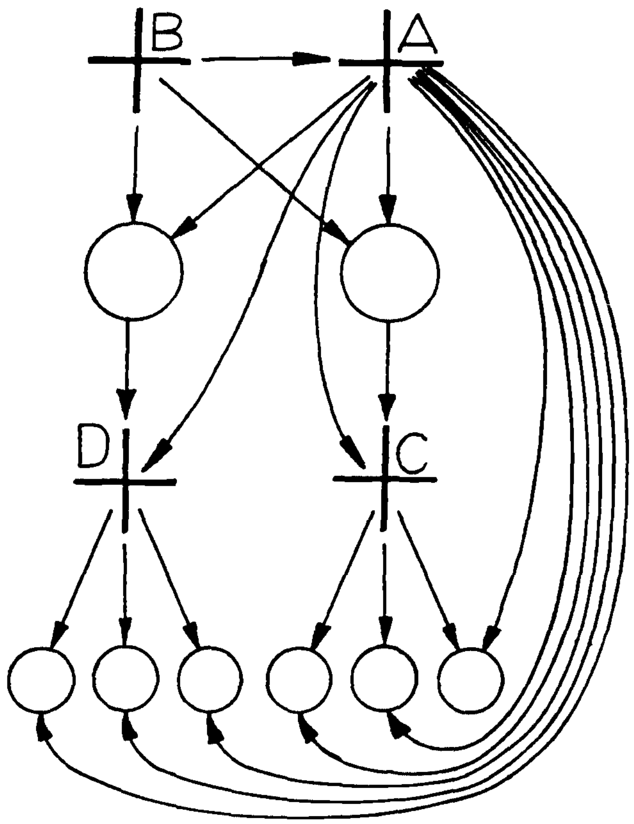


A part with two large holes and six small holes. Position tolerances are indicated by boxes containing the size of the tolerance and letters to indicate the datums relative to which the feature is positioned. The order of the letters shows the ordering of the datum-system constructed from these datums. The large holes are located relative to the sides A and B of the plate and the small holes are located relative to their nearest large hole and side A of the plate.

Figure 5.1.1

to one of the zones must be correct relative to the angle of A. The other three small holes are positioned similarly but relative to datums D and A.

A network can be created as shown in figure 5.1.2 with circle nodes representing features and cross nodes representing datums and directed arcs to show which items (features and datums) define which other items. Every datum is defined by a feature and every feature (except for the sides of the plate) is located relative to one or more datums. The six nodes at the bottom represent the small holes. All other nodes are annotated. Note that, in this network circle nodes represent features. In zone-datum networks features with more than one tolerance are represented by having a circle node to represent each zone.

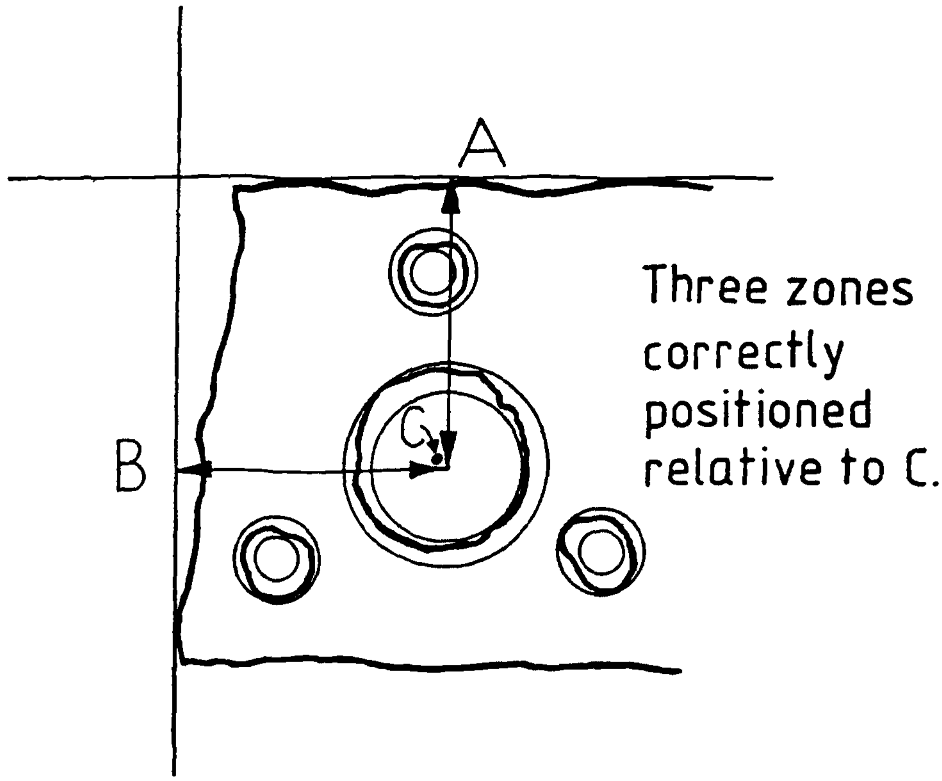


The network of features and datums for the part in figure 5.1.1.

Figure 5.1.2

Datums and features are treated as separate entities in the network. This is consistent with Requicha's formalism but is not made clear in an engineering drawing.

As a first step towards understanding the nature of the arcs in the network consider the exaggerated illustration of the real part in figure 5.1.3. As a simplification only one set of holes



An exaggeration of a portion of the part in figure 5.1.1. The large tolerance zone is correctly positioned with respect to A and B. The small tolerance zones are correctly positioned with respect to C.

Figure 5.1.3

is shown. Datums and tolerance zones are included and the actual surfaces of the part are shown by bold lines. Using this illustration it will be shown how there are constraints on the positions of datums and tolerance zones.

Datums A and B both rest against their associated features. Datum A rests as closely as possible against its associated feature. Datum B is secondary, however, and so is constrained to be perpendicular to A and, in the illustration, only touches its associated feature at one end.

The tolerance zone of the large hole lies at a distance of 5 units from each of A and B since the nominal hole lies at 5 units from each of the nominal faces corresponding to the datums. The real surface of the hole satisfies the tolerance specification only if it lies in this zone.

The measuring procedure (section 4.2.3) is used on this surface to define datum C. The datum ends up near the centre of the tolerance zone. It is not exactly centred on the zone but lies within a distance from it determined by the thickness of the zone (figure 5.9.7, p145). The tolerance zones of the small holes all lie at the correct distance from datum C and their direction from C is correct relative to datum A. The actual surfaces of these holes lie in their respective zones.

In this example there are three types of relationship between datums and tolerance zones.

- Firstly, datums in a datum system are correctly positioned with respect to one another.
- Secondly, tolerance zones are correctly positioned with respect to specified datums.
- Thirdly, datums are centred on their associated features and so are approximately aligned with the corresponding tolerance zones.

A fourth type of relationship would occur if features with more than one type of tolerance had been included. Each relationship puts constraints on the positions of tolerance zones and datums. Before discussing them further there are several topics that need to be formalised.

5.2. REPRESENTATION OF NOMINAL FEATURES, TOLERANCE ZONES AND DATUMS

The computational representation of a toleranced part consists of three types of geometric entity. These are,

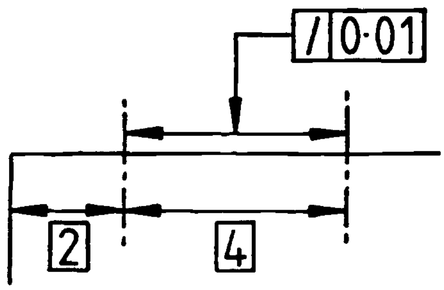
1. A decomposition of the surface of the part into **nominal features**;
2. **Tolerance zones** associated with nominal features;
3. **Datums** and **systems of datums**.

Requicha defined all of these but the following will describe what is required if they are to be represented computationally.

Representation of Nominal Features

Definitions of simple and composite features were given in chapter 4. The description of nominal features in a tolerance specification has simple features which are simple geometric surfaces and composite features which are sets of simple features.

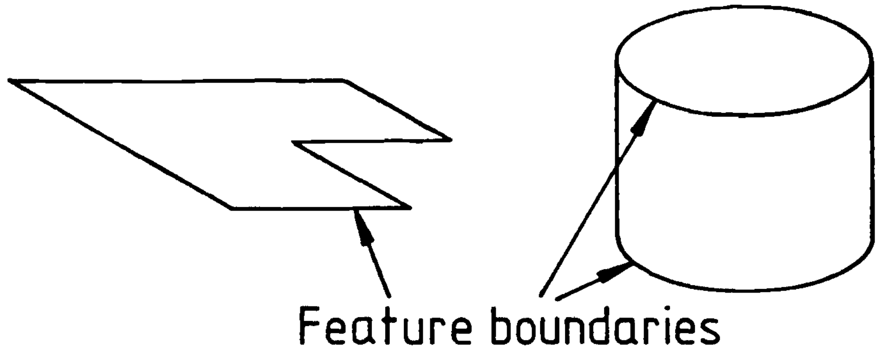
Note that a simple geometric surface might be split into more than one simple feature. This would enable a tolerance to be applied to only part of a surface and is conventionally indicated on a drawing as in figure 5.2.1.



A tolerance of flatness applied to only part of a surface.

Figure 5.2.1

A simple nominal feature can be represented by a description of the shape and position of its surface plus a description of the boundary of the surface. The boundary defines the extent of the feature. Figure 5.2.2 illustrates the boundary of a plane feature and of a cylindrical feature. A nominal feature has a coordinate system attached to it and it will be assumed that for planes and cylinders the coordinate system will be positioned as described in section 3.1 under "Features".



Boundaries of a planar feature and a cylindrical feature.

Figure 5.2.2

Representation of Tolerance Zones

The second type of geometric entity in the tolerance specification is the tolerance zone. These were formalised in section 4.2.4. They define a volume in which the real surface must lie. The representation of a tolerance zone has four components:

1. A pointer to the associated nominal feature (simple or composite).
2. A name, indicating the type of tolerance (form, size, position, orientation).
3. One or two tolerance parameters. The meaning of these depends on the type of tolerance but usually they define the thickness of the zone.

4. In the case of position and orientation tolerances there are pointers to datums or datum-systems.

The constraints on the shape and position of a zone can be deduced from this information if Requicha's formalism is followed.

Zones can be one of four types: form, size, orientation or position. Tolerance parameters are assumed to satisfy the constraints shown in figure 4.3.3. An MMC-position tolerance with a size tolerance can be converted to a position tolerance as described in section 4.3 and can, therefore, be specified with this representation.

Representation of Datums

The third type of geometric entity is the datum. Datums were formalised in section 4.2.3. A simple datum is a point, an infinite line or an infinite plane and may be easily represented. It has a coordinate system and a pointer to a simple or composite feature. A datum-system is a set of simple datums which may or may not be ordered.

5.3. COMPUTATIONAL REPRESENTATIONS AND FORMALISMS

The set of allowable shapes of a part is called its variational class. The formalism presented in the last chapter along with a tolerance specification for a part is a mathematical description of a variational class. In the current chapter it is shown how the variational class can be described in a computer. This description will be referred to as a "computational representation" or simply a "representation".

The validity of a computational representation can be checked by determining if it can be used to answer certain questions correctly. Questions asked in the real world are **measurements** and questions asked about the representation are **calculations** or simulated measurements. A similar scheme is suggested by Requicha (1977). Note that whilst a measurement performed on a single instance of a part returns a single value a calculation returns a set of values. This is because the calculation operates on a description of the complete set of possible parts and returns the set of all possible values that could be obtained from the measurement.

A calculation may return values which do not correspond exactly with the measurement it simulates for two reasons: either the representation itself may be an approximation or the calculation may be an approximation.

Let V be the variational class of a part and let V' be a variational class defined by an inaccurate representation. Let m be a measurement so that $m(i)$ is the value obtained by the measurement if it is applied to an instance of the part denoted by i ($i \in V$). Let $m(V)$ be the set of all possible values of the measurement i.e. $\{m(i): i \in V\}$. Let m' be a calculation corresponding to m so that $m'(V')$ is the set of values obtained by applying m' to the computational representation of the variational class. Ideally, $m'(V')$ would equal $m(V)$ but it will be shown that it is acceptable if $m'(V')$ contains $m(V)$.

In the introduction to this chapter it was shown that any design requirement must hold for all parts in the variational class. Suppose that we require the result of a measurement, $m(i)$, to be within a certain range. Then it follows that all members of $m(V)$ must be in this range. Therefore, any design requirement, involving dimension, m, can be written in the form,

$$m(V) \text{ is contained in } M,$$

for some set M.

Assuming $m'(V')$ contains $m(V)$ we can deduce that $m(V)$ is contained in M knowing that $m'(V')$ is contained in M. Hence it is possible to test whether design requirements hold using an inaccurate representation.

The converse, that $m'(V')$ is not contained in M, does not imply that $m(V)$ is not contained in M. Therefore, we have only a partial decision procedure for deciding whether or not a tolerance specification is satisfactory: either the specification **is** satisfactory or it **might not** be satisfactory. Further discussion on such partial decision procedures can be found in Brooks (1983).

The question arises as to how much inaccuracy can be tolerated if useful results are to be obtained from the representation. A system that always returned the result that the requirements **might not** be satisfied would be useless even though it would always be correct. A system must be accurate enough to be useful.

It is not easy to define how accurate a system should be. One useful property, however, would be the following. Suppose that, in the real world, a part satisfies its design requirements when one of its dimensions is given a tolerance of T or less. It should be possible to predict from the representation that tightening the tolerance on that dimension does cause the requirements to be satisfied. However, a system might be acceptable if the required tightening on this particular tolerance was predicted to be some value T' less than T. A system would be unacceptable if it could not predict that tightening the tolerance caused the design requirements to be satisfied.

5.4. VARIATIONAL CLASSES AND ZONE-DATUM STRUCTURES

A description of a variational class in the computational representation has the form of a set of tolerance zones and datums linked by constraints. No direct mention is made in the representation of real surfaces or of the shapes of real parts. The following is a discussion on how the set of zones and datums with constrained positions relates to the variational class of a part.

Every toleranced part has a variational class as defined in section 4.2.1, which is a set of regions of three-dimensional space (R^3) which copies of the part can occupy. A member of a variational class will be called an instance of the part and the symbol i will usually denote this.

The positions of the zones and datums of a part are constrained relative to one another in ways that will be described later in this chapter. A "configuration" of a set of n zones and datums will mean an n-tuple each component of which represents the position of one of the zones or datums. Each component can be represented as an ordered list of DOF-variables. The set of zones and datums in a definite configuration can be thought of as a rigid structure. It will be called a zone-datum structure or ZDS for short. There is a set of such configurations for which the constraints on the positions of zones and datums are satisfied and this will be called the ZDS set. The symbol Z will usually be used to denote a ZDS set and a member of Z will usually be denoted by I.

Hence, the representation describes a set of possible zone-datum structures. However, the variational class of a part is a set of regions of space. The link between these two sets is described in the following paragraph.

Suppose V is a variational class for a part and Z is its ZDS set. For each member of Z there is an equivalent subset of V. Therefore, a mapping can be defined from Z to the set of subsets of V. Call this mapping σ. Given a zone-datum structure, I in Z, $\sigma(I)$ is the subset of V representing instances of the part whose surfaces are contained in the zones of I.

Conversely, but not so importantly, there is a mapping from V to the set of subsets of Z such that for all i in Z and for all I in $\sigma'(i)$, $\sigma(I)$ contains i. In other words $\sigma'(i)$ is the set of zone-datum structures that could give rise to i.

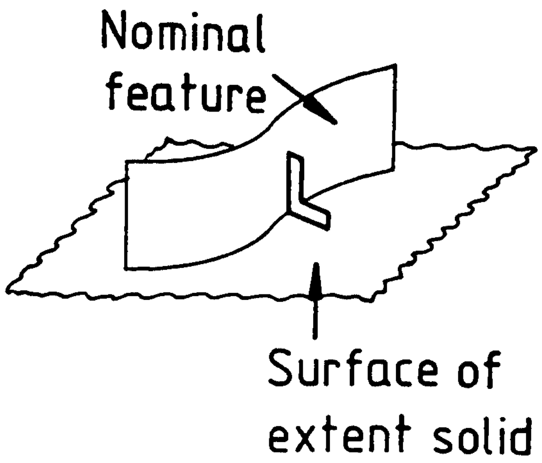
5.5. EXTENT-SOLIDS OF FEATURES

Informally, an <u>extent-solid</u> is a solid which is bounded in the same way as a nominal feature but extends infinitely in directions along normals to the nominal feature. They allow the extent of a feature to be specified in a convenient way. They are used for isolating the portions of tolerance zones, datums and real features which correspond to the extent of the nominal feature. Figure 5.5.2, to be explained further below, shows some features and their extent-solids.

The extent-solid of a feature has the following properties.

1. It contains the nominal feature.
2. Its surface contains the boundary of the nominal feature.
3. Where its surface intersects the boundary of the nominal feature its surface is normal to the nominal feature (figure 5.5.1).

The formal definition of an extent-solid first requires the definition of an <u>offset surface</u> of a feature. Let F be a nominal feature consisting of simple features and let H be its extended



The surface of an extent-solid must be normal to the nominal feature where the two meet.

Figure 5.5.1

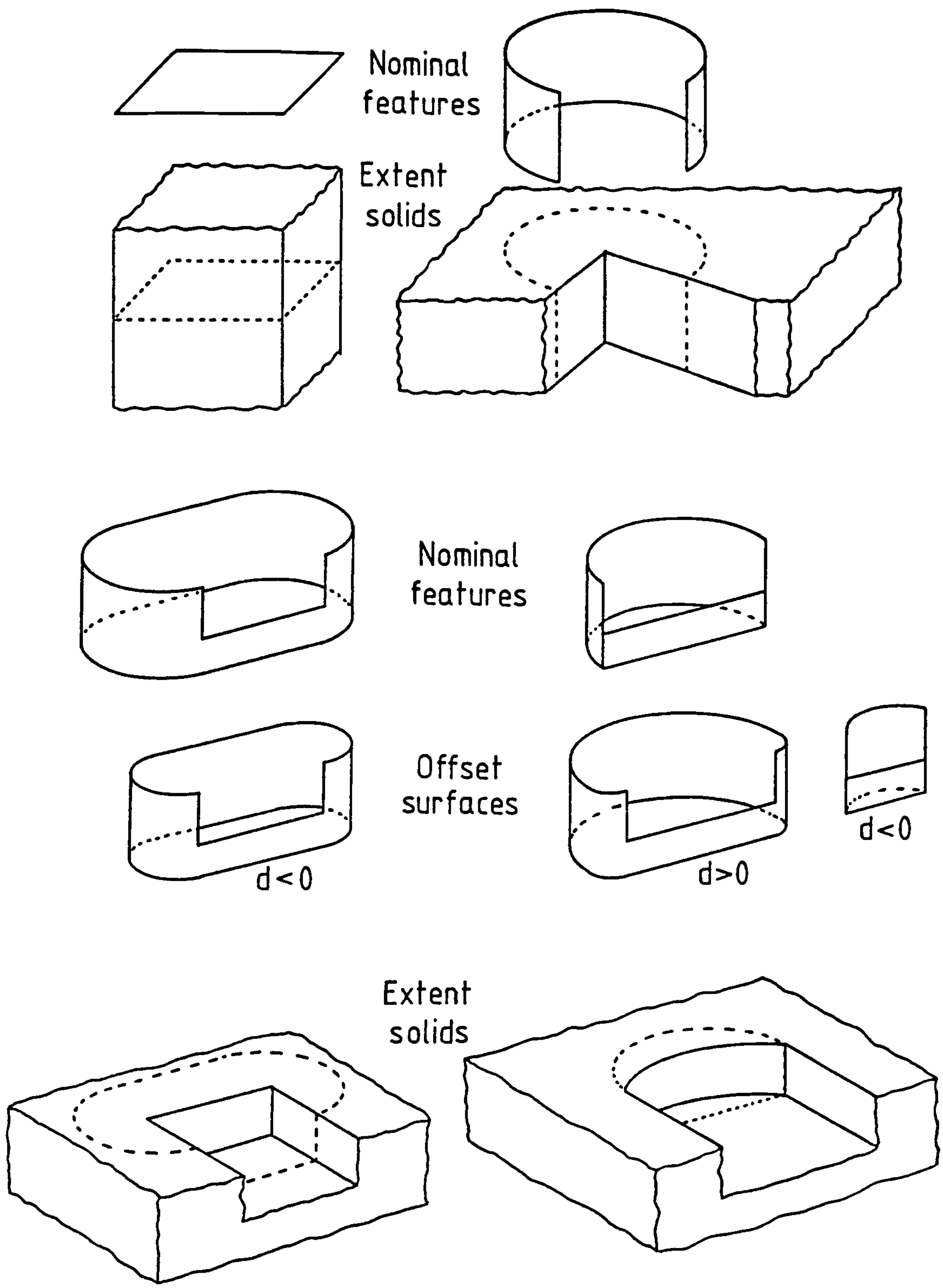
feature. Let d be a number. Then the offset surface of F is defined as the result of projecting F onto the surface of O(H;d) or more precisely as

$$\{p \in \partial O(d;H): \text{there exists a point on F whose distance from } \mathbf{p} \text{ is d}\}.$$

Note that for many features the offset solid is empty when d is below a certain (negative) value. For such values of d the offset surface is undefined.

The extent-solid is defined as the set of all points which lie on an offset surface of the feature for any offset for which the offset surface is defined. It is the volume swept out by the offset solid as the offset increases to infinity and decreases to negative infinity or to the minimum value for which the offset solid is defined.

Figure 5.5.2 shows extent-solids for several features and includes offset surfaces for two composite features.



Nominal features and their corresponding extent-solids.
Above, a planar feature and a cylindrical feature.
Below, composite features, including examples of offset surfaces.

Figure 5.5.2

The extent-solid can be used to define the significant portion of a tolerance zone. This is obtained by intersecting the zone with the extent-solid of the feature. Although most tolerance zones have infinite extent, a real feature only "occupies" a finite portion of a zone. The zone can be cut down to a finite extent equal to the extent of the nominal feature without significantly affecting the class of real features which it can contain. A real feature might extend slightly beyond the significant portion of the zone. However, the amount that it extends will be much less than the nominal size of the feature and is therefore not important.

5.6. SIGNED DISTANCES

In section 3.4 the concept of a signed distance was introduced for finding constraints between features. In this section signed distances are formalised using offset solids.

Let A and B be two regions of space. Let d be the minimum value for which A intersects O(d;B). Then the signed distance between A and B (denoted sdist(A,B)) is defined to be d.

Note that if A and B intersect then the minimum value of d for which A and O(d;B) intersect is negative. However, if A and B do not intersect then d must be positive. In this case d is the straightforward distance between A and B. This can be seen as follows. Let the distance between A and B be d so that there exists a point p in A whose distance from B is d. Then p is a member of O(d;B). Hence, A intersects O(d;B). We also need to show that A does not intersect O(d′;B) for d′<d. If A did intersect O(d′;B) then a contradiction would arise since there would be a point in A whose distance from B was less than or equal to d′. It follows that the distance between A and B is the minimum value of d for which A and O(d;B) intersect. Hence, according to the definition above, the signed-distance between A

and B is d.

Often we will be interested in the signed distance between two objects, A and B, with variable position. In this case sdist(A,B) converts to an expression involving the position variables of the objects.

This allows certain geometric constraints to be converted to an algebraic form. For example, the condition that A and B do not intersect can be converted to the algebraic constraint,

$$\mathrm{sdist}(A,B) > 0.$$

If the interiors of A and B do not intersect then

$$\mathrm{sdist}(A,B) \geq 0.$$

In this thesis the boundaries of volumes are not usually important so these two conditions are not usually differentiated. The condition that A is contained in B can be written,

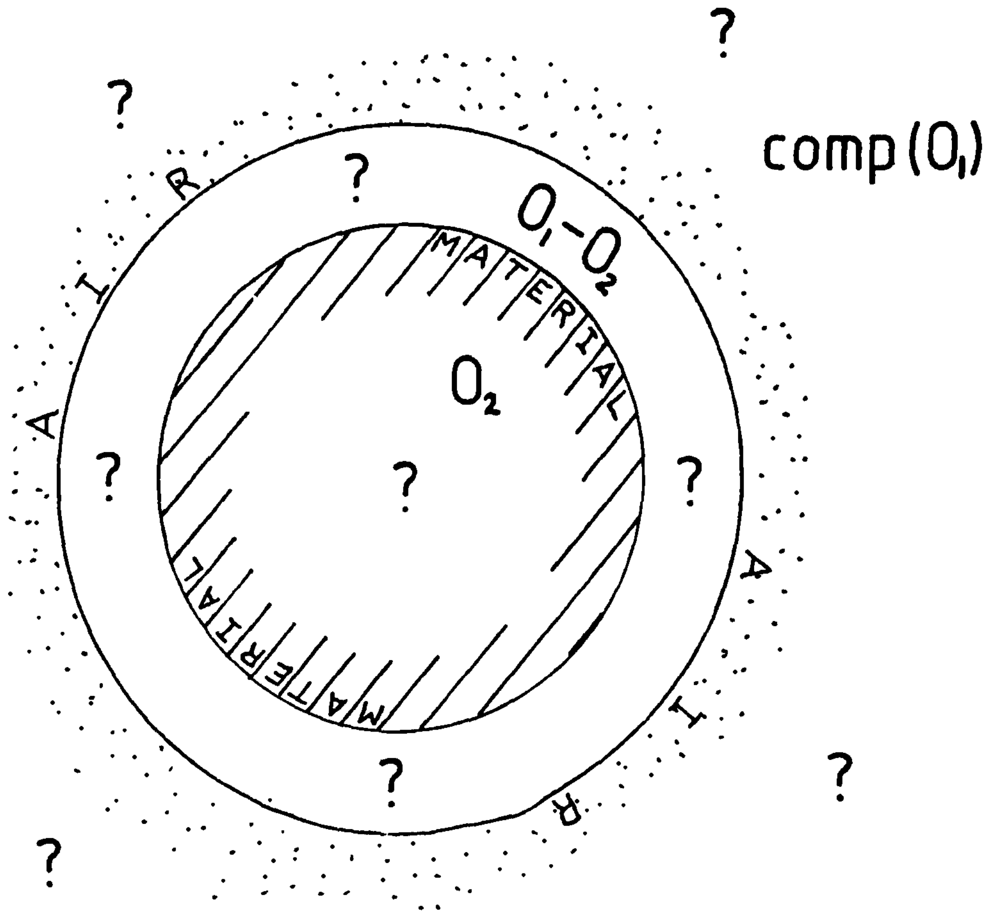
$$\mathrm{sdist}(A,\mathrm{comp}(B)) \geq 0,$$

where "comp" represents set complement.

Appendix 1 explains how signed distances can be evaluated in the particular situations where they are required later in this thesis.

5.7. THE DISTRIBUTION OF AIR AND MATERIAL IN A ZONE

A zone expressed as O_1-O_2, where O_1 and O_2 are offset solids of an extended feature, divides space into three disjoint regions, comp(O_1), O_1-O_2 and O_2. Let E be the extent-solid of the zone's feature and consider the distribution of air and material in comp(O_1)nE, (O_1-O_2)nE and O_2nE (figure 5.7.1). The symbol "n" is used to represent set intersection. This will be useful for deriving constraints in relationships involving zones in section 5.9 and in chapter 6.



A zone splits space into three regions. In the significant portion of the zone material and air can be guaranteed to lie close to the within and outside surfaces of the zone. Areas containing question marks - inside the zone and far away from the zone - can contain either air or material.

Figure 5.7.1

To a good approximation the real feature is contained in (O_1-O_2)nE with one side of the real feature being air and the other being material. It follows that every point of (O_1-O_2)nE could be either air or material.

Points in $O_2 \cap E$, on the other hand, are guaranteed to lie in material if they are close to the surface of O_2. Conversely, points in comp(O_1)$\cap E$ are guaranteed to lie in air if they are close to the surface of O_2. Thus there are layers of air and material surrounding the zone.

The thickness of the layers of air and material depends on the presence of other features. However, a feature is often considered independently of all others and then it is unimportant what happens far away from the surfaces of O_1 and O_2. In this case it is convenient to consider that the region, $O_2 \cap E$, is entirely material and that the region, comp(O_1)$\cap E$, is entirely air. In the future this assumption will sometimes be made without comment. The regions will be referred to as the <u>air</u> and <u>material regions of the zone</u>.

5.8. POSITIONS OF TOLERANCE ZONES AND DATUMS

This section describes how the positions of zones and datums can be represented. The next section will show how constraints arise on their positions and so a representation is needed for position constraints. Basically, the method used for defining the positions of tolerance zones and datums is the same as for defining the positions of parts in chapter 3. Variations in the positions of zones and datums are assumed to be very small compared to the size of features.

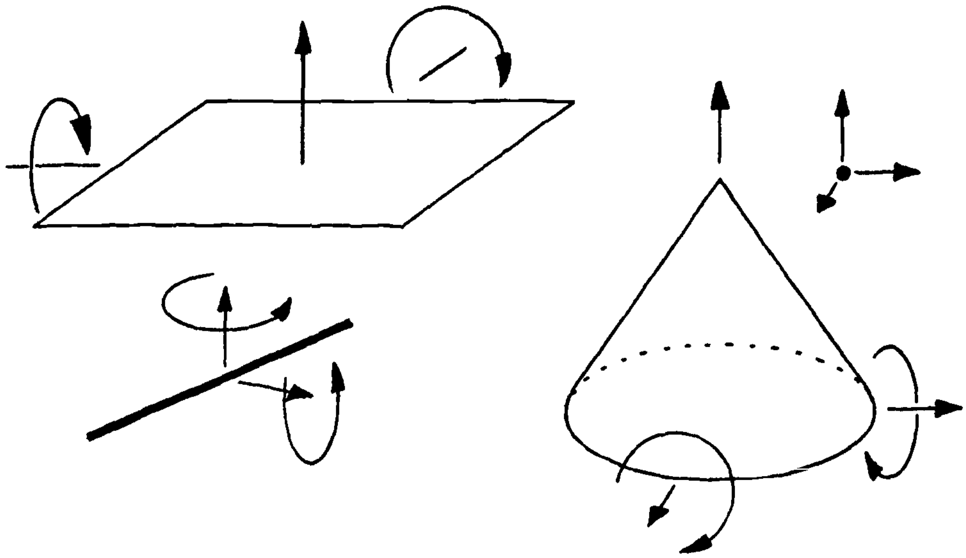
All zones and datums have a coordinate system attached to them. The position of both types of item (zones and datums) is variable and so is not necessarily the same as the position of the corresponding nominal feature. A position is measured relative to its nominal value and can be expressed as the rotational and translational displacements that are necessary to arrive at the actual position from the nominal position. The coordinate system is used to define the three rotations and three translations and a variable is associated with each. The variables are called DOF-variables and will be denoted δx, δy, δz, $\delta\theta$, $\delta\phi$ and $\delta\psi$. Rotations θ, ϕ and ψ are about the x, y and z-axes of the coordinate system respectively (figure 3.3.1, p55). They are analogous to the DOF-variables used in chapter 3 to represent variations in positions of parts. However, note that capital Δ's were used in chapter 3 whereas small δ's are used here. In chapter 6 both types of DOF-variable will be used and the different notation will indicate variations in the shape of parts and variations in the positions of parts.

A set of DOF-variables will be associated with each relationship between zones and datums. Each relationship has its own set of DOF-variables. However, for simplicity the set of symbols given in the last paragraph will be used when no confusion can arise. Where it is necessary to differentiate between DOF-variables from different relationships the relationships will be numbered and the DOF-variables will be subscripted with the relevant number.

The constraints to be derived in the next section will be expressed as equalities or inequalities involving DOF-variables.

Note that most zones and datums have less than six degrees of freedom because many motions map the items onto themselves. Therefore, these motions can be ignored. Figure 5.8.1 shows examples of zones and datums with their degrees of freedom; an infinite plane has two rotations and one translation; an infinite line has two translations and two rotations; an infinite cone has two rotations and three translations; a point has three translations and no rotations. The motions which map an entity onto itself will be called redundant.

Sometimes, it is convenient to define the nominal position of an item in terms of all of its degrees of freedom despite some



Infinite datums and zones have less than six degrees of freedom. Arrows indicate the degrees of freedom that they do have.

Figure 5.8.1

being redundant. A coordinate system attached to the item makes this possible since the position of a coordinate system is affected by all degrees of freedom.

5.9. RELATIONSHIPS BETWEEN ZONES AND DATUMS

A part is to be represented as a zone-datum network which has zones and datums as nodes linked by arcs to represent relationships between them. Each arc joins two zones or datums which have some geometric relationship between them. The geometric relationship can be converted to a set of inequalities which are then attached to the arc.

The zone-datum network has many similarities to the network of parts and relationships in chapter 3. Each relationship can be analysed in isolation from all other relationships. This is because a relationship implies constraints on the positions of just two zones or datums and says nothing about any other zones or datums. When constraints have been attached to each relationship then the network can be analysed in a manner similar to the network of chapter 3.

There are several ways in which geometric relationships between zones and datums occur.

1. The position of a zone of position tolerance has a fixed position relative to some specified datum.
2. The relative position of the datums in a datum system are fixed.
3. The relative position of tolerance zones associated with the same feature are constrained. This is because they must intersect over a large enough volume to contain the real feature.
4. The position of a datum is constrained, via the measuring procedure relative to the zones of its associated feature.

Numbers 1, 2 and 4 were introduced in the example of section 5.1 and number 3 is mentioned for the first time here. All four ways that relationships occur will be discussed in detail in this section. Each gives rise to a geometric constraint which is converted to an algebraic constraint in the form of a set of inequalities. This section only deals with simple features: composite features are left until section 5.10.

5.9.1. Relationships Locating Zones Relative to Datums

This type of relationship occurs when a zone of position or orientation tolerance has fixed location or orientation relative to a datum. In the example of section 5.1 the tolerance zones of the large holes were positioned by datums A and B. Therefore, each of these zones of position tolerance makes a relationship with both of A and B.

Each relationship removes some degrees of freedom from the relative position of the datum and the zone. Since fewer than six parameters are usually required to define the position of a datum or a zone it follows that the number of degrees of freedom constrained in the relationship will in general be less than six. However, it is sometimes useful to put constraints on some of the redundant degrees of freedom. This is possible if the zone has a coordinate system attached to it.

For example, consider figure 5.9.1(i) showing a part with one small hole and two large holes. The large holes define a datum system, AB, which is used to locate the small hole. Datum A is primary and datum B is secondary. Therefore, A and B are parallel and are separated by the same distance that the axes of the nominal features are separated.

The axis of the tolerance zone of the small hole must lie on the plane containing the two datums. Unfortunately, this is a geometric relationship between three items and so is not directly representable in the zone-datum network where all relationships act between pairs of items. Therefore, it is necessary to find a way of expressing the position of the small hole relative to each of A and B independently.

The position of the tolerance zone of the small hole lies at a fixed distance from each of the datums. This can be represented as two relationships each between the tolerance zone and one of the datums. Each relationship constrains the zone to lie on a circle centered on the datum as shown in figure 5.9.1(ii). The intersection of these occurs at a single point and so fixes the position of the zone.

Note that the constraints here are non-linear. Under the assumption that tolerances are small, however, it is possible to linearise constraints in most circumstances without causing major inaccuracies. However, in the situation described here problems do arise.

Linearising the constraints means that each relationship constrains the axis of the zone to be parallel with the datums and to lie on a plane as shown in figure 5.9.1(iii). Algebraically the constraints are

$$\delta z=0\ ,\quad \delta\phi=0\ ,\quad \delta\psi=0.$$

(Note that, in general, DOF-variables are equated with zero in this type of relationship because zero is the nominal value of a DOF-variable.) The problem is that, each relationship now implies the same constraints and so the union of both sets of constraints is the same as a single set of constraints. Hence the position of the zone is not uniquely determined.

It is possible to overcome this problem by attaching a coordinate system to the zone (as in figure 5.9.1(i)). It can now be said that the zone (1) lies at a fixed distance from each datum and (2) has its y-axis colinear with each datum. The constraints arising from these two conditions do uniquely define the position of the zone even when linear approximations are made.

The constraints imposed by the datums on the zone are now (with linear approximation)

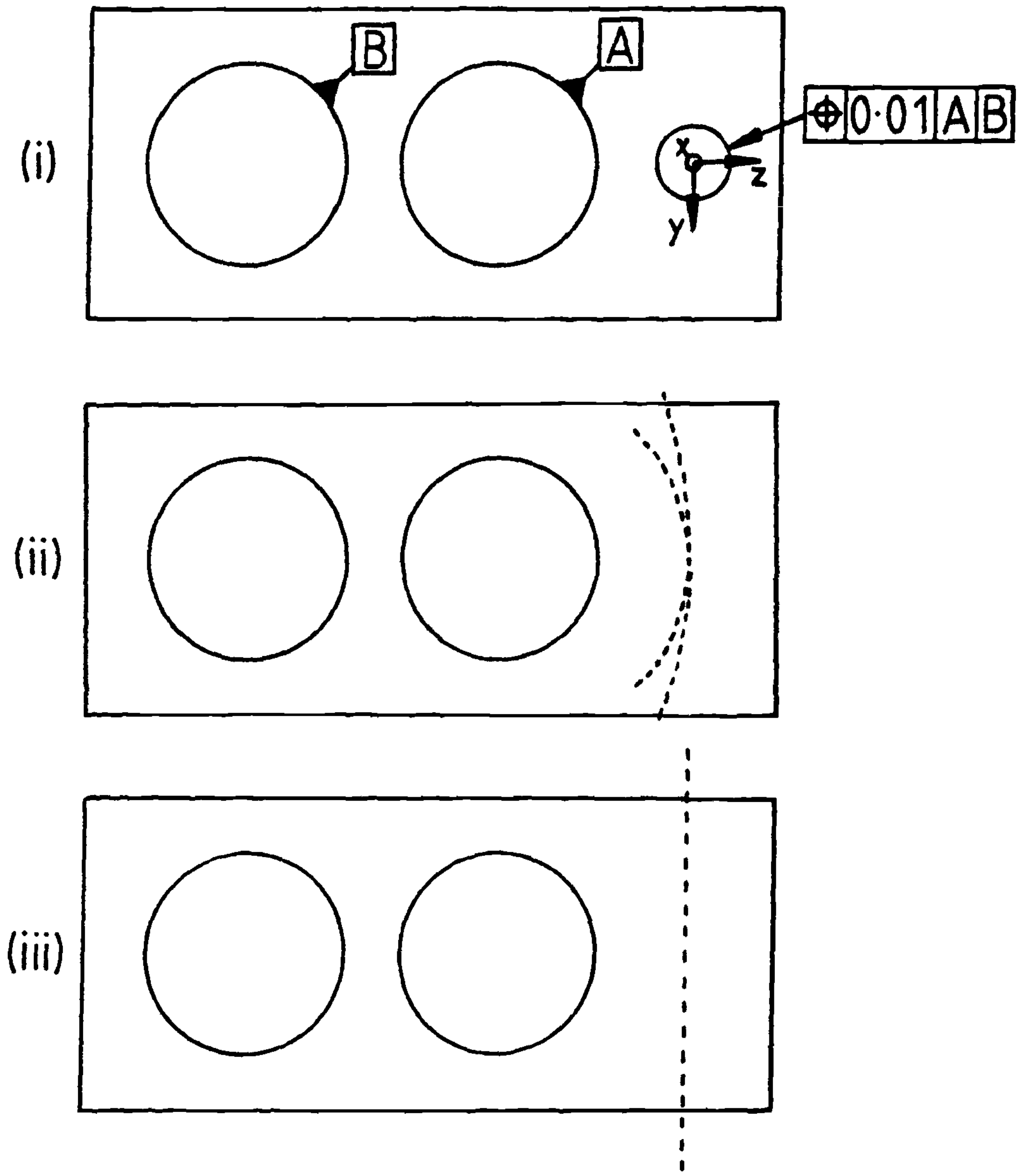
$$\text{Datum A:}\quad \delta y-a\delta\theta=0\ ,\quad \delta z=0\ ,\quad \delta\phi=0\ ,\quad \delta\psi=0,$$

$$\text{Datum B:}\quad \delta y-b\delta\theta=0\ ,\quad \delta z=0\ ,\quad \delta\phi=0\ ,\quad \delta\psi=0,$$

where a and b are the distances between the axis of the respective datum and the axis of the zone. Together these imply that

$$\delta y=0\ ,\quad \delta z=0\ ,\quad \delta\theta=0\ ,\quad \delta\phi=0\ ,\quad \delta\psi=0.$$

These are strong enough to uniquely determine the position of the zone. The redundant degree of freedom $\delta\theta$ has provided constraints due to its interaction with non-redundant degree of freedom δy.



The small hole, in figure (i), is positioned relative to the two datums defined by the large holes. Each datum constrains the tolerance zone of the small hole to lie on one of the arcs in figure (ii). However, linearisation of the constraints means that each datum constrains the zone to lie on the plane shown in (iii). Thus its position is no longer uniquely defined.

Figure 5.9.1

To summarise, the position of a tolerance zone should be expressed relative to each datum individually because each relationship involves only two zones or datums. Linearising constraints can mean that they no longer have a unique solution but attaching a coordinate system to the zone means that stronger conditions can be obtained which do give a unique solution.

Another problem is the choice of suitable coordinate systems relative to which to derive the constraints. It is often convenient to derive the constraints relative to coordinate systems other than the main coordinate systems of the zones or datums. This ensures that the constraints take a standard form. Then, by making suitable variable substitutions the constraints can be converted to the main coordinate systems. Convenient coordinate systems should be chosen to be coincident with the coordinate systems of the datums.

For example, consider the constraints on a zone relative to a linear datum with a coordinate system attached to the datum with x-axis colinear with the datum. The constraints take the following standard form with respect to this coordinate system:

$$\delta y=0 \ , \quad \delta z=0 \ , \quad \delta\phi=0 \ , \quad \delta\psi=0.$$

Note that in general the constraints from this type of relationship equate DOF-variables or expressions of DOF-variables to zero. This is because zero is the nominal value of a DOF-variable.

Suppose the zone is located 5 units from the datum in the direction of the y-axis. DOF-variables associated with the zone's coordinate system will be denoted with a dash. They are related to the DOF-variables of the datum's coordinate system by

$$\delta x=\delta x'+5\delta\psi', \quad \delta y=\delta y', \quad \delta z=\delta z'+5\delta\theta', \quad \delta\theta=\delta\theta', \quad \delta\phi=\delta\phi', \quad \delta\psi=\delta\psi'.$$

Hence, the constraints on the zone in terms relative to its own coordinate system are

$$\delta y'=0 \ , \quad \delta z'+5\delta\theta'=0 \ , \quad \delta\phi'=0 \ , \quad \delta\psi'=0.$$

Constraints relative to the coordinate system of a datum take a standard form. These can be converted to the zone's coordinate system by suitable change of variable. This technique of choosing a convenient coordinate system so that constraints initially have a standard form is also useful with other types of relationship.

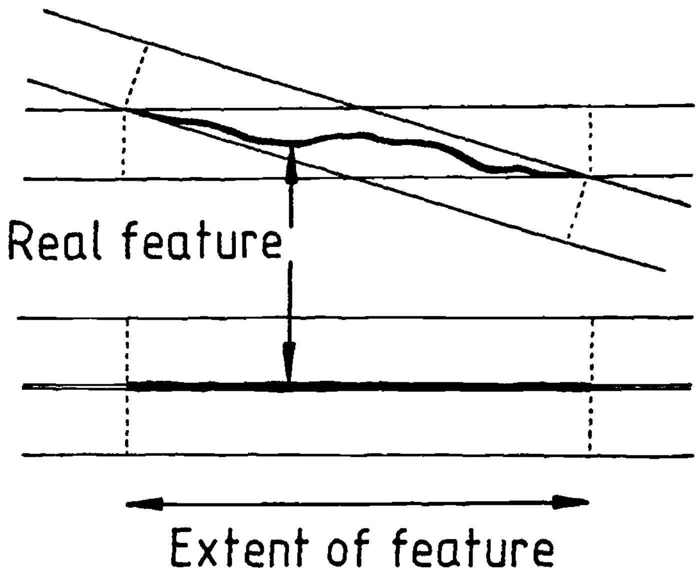
5.9.2. Relationships Between Datums in a Datum-System

In general a datum-system is a partially ordered set of datums. Datums in ordered datum-systems have fixed positions relative to one another according to the relative positions of their associated nominal features. In the example of section 5.1 the two planar datums are perpendicular because their associated features are perpendicular. The angle between the datums is fixed and so rotation of the secondary datum about an axis parallel with the line of intersection of the datums is constrained. However, all other degrees of freedom are unconstrained. As with the previous relationship type the constraints equate DOF-variables with zero. In this example the constraint is

$$\delta\phi=0.$$

The degrees of freedom which are constrained depend on the types of datums involved and the disposition of the datums. For example, a datum-system consisting of two perpendicular intersecting line datums (x-axes colinear with the datums) would give the same constraint as above. However, a datum-system consisting of two parallel line datums, such that their y-axes lie in the common plane of the datums, would have the constraints,

$$\delta y=0, \quad \delta\phi=0, \quad \delta\psi=0.$$



Above, extreme relative inclination and, below, extreme translational displacements of two zones, associated with the same planar feature, given that a real feature has to fit in their intersection.

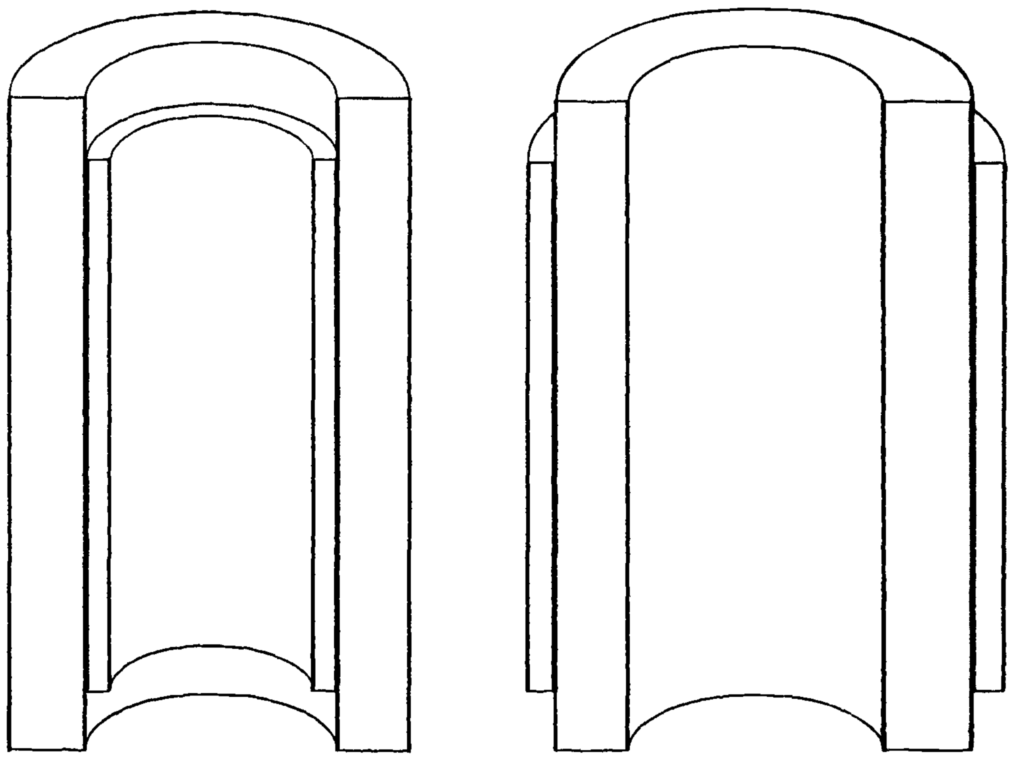
Figure 5.9.2

5.9.3. Zones Associated with a Single Feature

The previous two relationship types involved equality constraints but the following two involve inequalities. The extent of a feature is important in these relationships. Although the extent of a real feature varies between instances of the part the approximation is made that it is the same as the extent of the nominal feature. Use will be made of the "significant portions" of zones (see end of section 5.5).

This section discusses the constraints on the positions of zones associated with a single feature. Although size and form tolerance zones have position which is completely undefined by the formalism their positions are constrained by a simple geometrical consideration: there must be room in the intersection of the zones for a real surface. The real feature has as extent which is approximately the same as the nominal feature.

Consider a planar feature with a form tolerance and a position tolerance. Figure 5.9.2 shows, firstly, two zones of planar features at their extreme inclination given that a surface with the required extent has to fit inside them. Any larger inclination would cause the length of the region of intersection to be shortened. Secondly, figure 5.9.2 shows the most extreme translational displacement that can occur if the zones are to intersect.



The maximum and minimum size of a form tolerance zone (lesser thickness) relative to a position tolerance zone (greater thickness). In these situations the real feature is forced to lie in the plane of intersection of the zones.

Figure 5.9.3

Now consider a cylinder with a form tolerance and a position tolerance. The size of the form-tolerance zone is undefined by the formalism but is constrained by having to intersect the position-tolerance zone. Figure 5.9.3 shows the minimum and maximum sizes of a cylindrical form tolerance zone that are attainable. Extremes of inclination and translational displacement occur in a similar way to figures 5.9.2(i) and (ii).

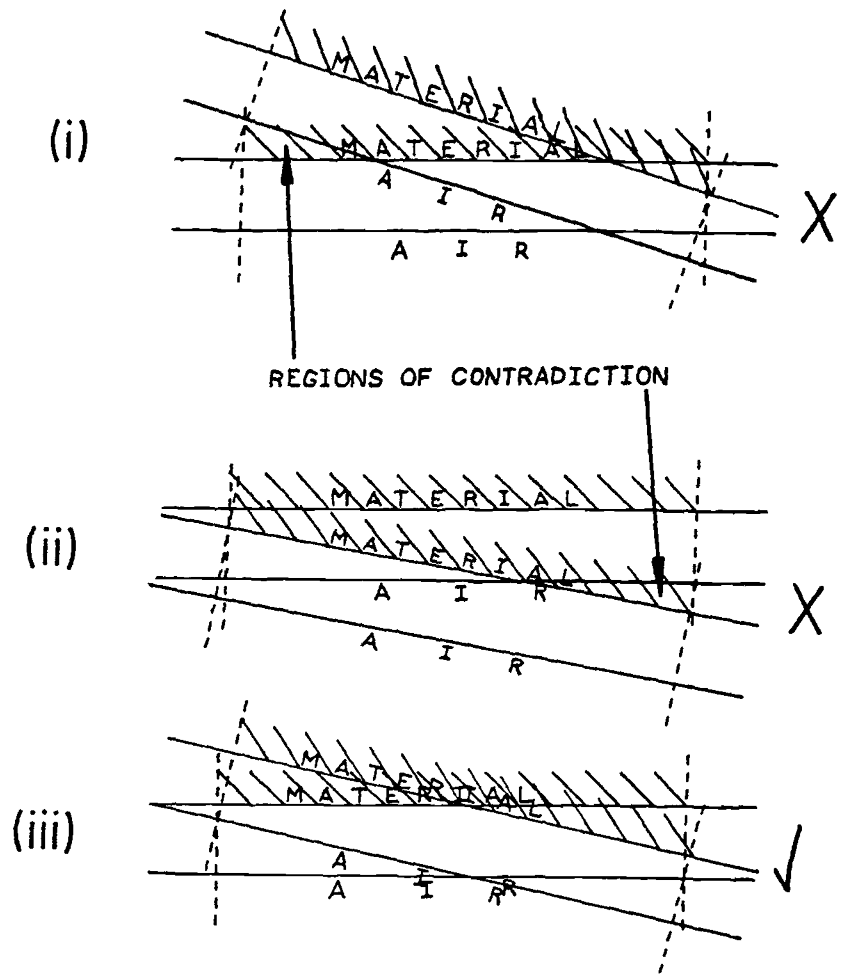
The geometric constraint on the relative position of two zones can be expressed in terms of air and material regions of the zones. Recall from section 5.7 that each zone has a material region and an air region. If two zones are associated with the same real feature then the material region of one zone must not intersect the air region of the other zone. This condition is illustrated in figure 5.9.3A. In figures 5.9.3A(i) and (ii) the air region of one zone does intersect the material region of the other. It can be seen that this results in the zones intersecting over a length which is shorter than the extent of the feature. In figure 5.9.3A(iii), however, there is no contradiction since air regions do not intersect material regions.

Note that nothing has been said about the boundaries of air and material regions. It will be assumed that the above condition applies only to the interiors of air and material regions. Thus the interiors of the air and material regions do not intersect. This means that the resulting algebraic constraints contain "≤" and "≥" instead of "<" and ">".

Suppose that a feature has two tolerances with zones defined by O_1-O_2 and o_1-o_2 where O_1, O_2, o_1 and o_2 are all offset solids. Suppose the extent-solid of the feature is E. Then the material regions are O_2nE and o_2nE and the air regions are $comp(O_1)nE$ and $comp(o_1)nE$. The condition that two regions do not intersect can be expressed by saying that the signed distance between them is greater than zero. Hence the non-intersection of air and material regions can be expressed algebraically as,

$$sdist(O_2nE , comp(o_1)nE) \geq 0 \quad \text{and}$$

$$sdist(o_2nE , comp(O_1)nE) \geq 0.$$



Constraints between air and material regions of two zones associated with the same feature. In (i) and (ii) there is a contradiction since air regions and material regions intersect. In (iii) there is no contradiction.

Figure 5.9.3A

These convert to inequalities involving DOF-variables of the zones. If one of the zones is of form tolerance then there will be an additional variable representing the size of this zone.

As an example consider a slot feature consisting of two parallel rectangular planes facing one another with nominal dimensions A by B and with nominal separation W (figure 5.9.4(i)). The extent-solid of the feature denoted by E is an infinitely long rectangular prism whose cross-section has dimensions A by B.

Suppose the feature has a position tolerance and a size tolerance defined, respectively, by zones,

$$O(P_1;H')-O(P_2;H') \quad \text{and} \quad O(S_1;H'')-O(S_2;H''),$$

where H′ and H″ are copies of the associated extended feature. Because H′ is used in the definition of a zone of position tolerance it must be correctly positioned relative to some datum-system.

The positions of the zones are constrained in terms of signed distances as follows:

$$\text{sdist}\big(\ \text{comp}(O(P_1;H'))\cap E\ ,\ O(S_2;H'')\ \big) \geq 0, \qquad \text{C1}$$

$$\text{sdist}\big(\ \text{comp}(O(S_1;H''))\cap E\ ,\ O(P_2;H')\ \big) \geq 0.$$

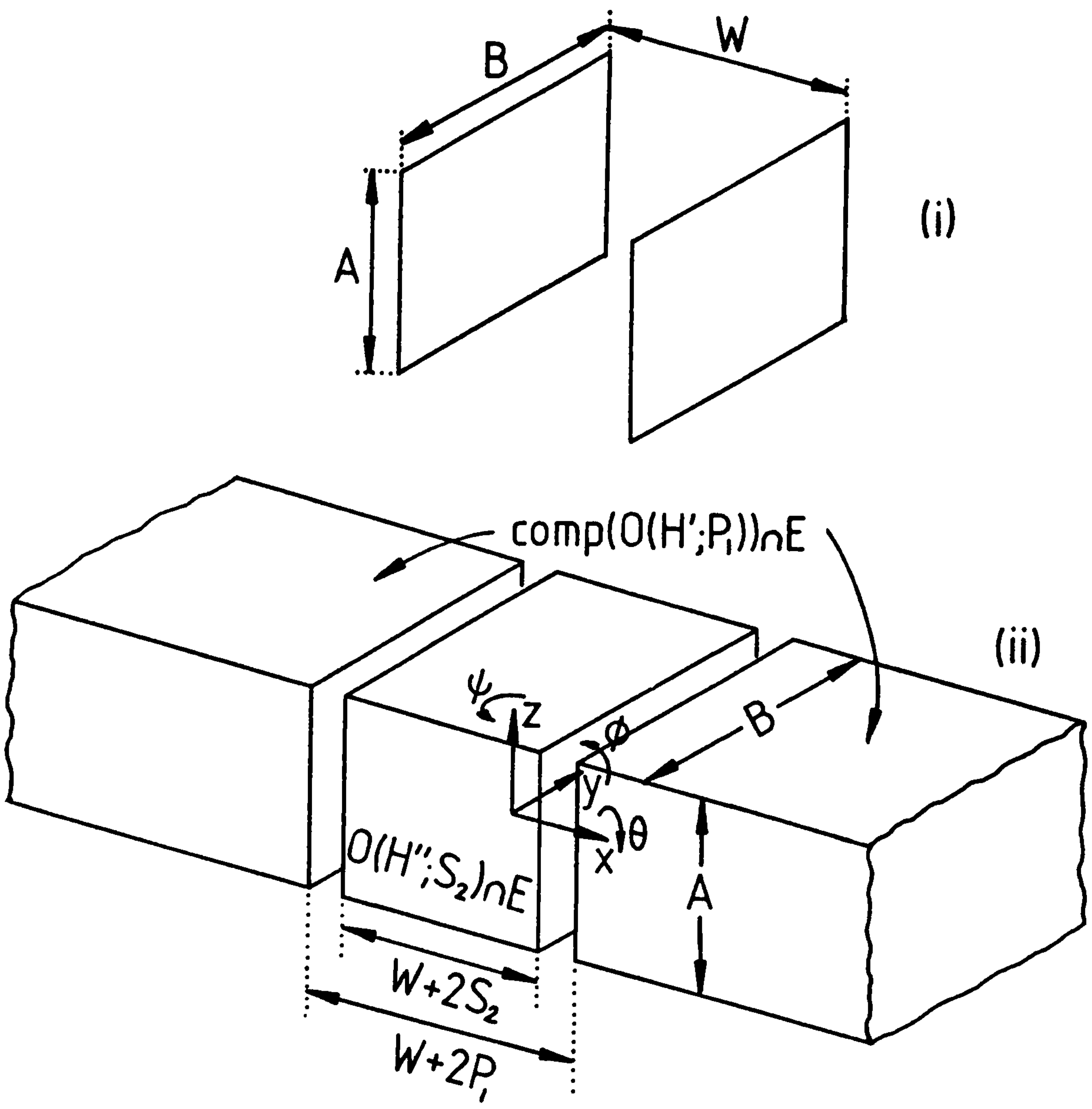
The solids occurring in the first of these expressions are shown in figure 5.9.4(ii). The solids in the second expression are similar except that the widths of the slot and the block are respectively $W+2S_1$ and $W+2P_2$. Each expression is the signed distance between a slot and a block fitting in the slot. Due to the small uncertainty assumption the two objects only come into contact between surfaces that correspond to the nominal feature. These are the faces that form the gaps between the solids in figure 5.9.4(ii). A general signed distance expression between such a slot of width W_1 and a block of width W_2 both with cross-section A by B is

$$(W_1-W_2)/2 - |\delta x| - (A/2)|\delta\phi| - (B/2)|\delta\psi|$$

where x, ϕ and ψ are directions and rotations indicated in figure 5.9.4(ii). Hence, the algebraic constraints on the positions of the two tolerance zones are

$$P_1-S_2 - |\delta x| - (A/2)|\delta\phi| - (B/2)|\delta\psi| \geq 0,$$

$$P_2-S_1 - |\delta x| - (A/2)|\delta\phi| - (B/2)|\delta\psi| \geq 0.$$

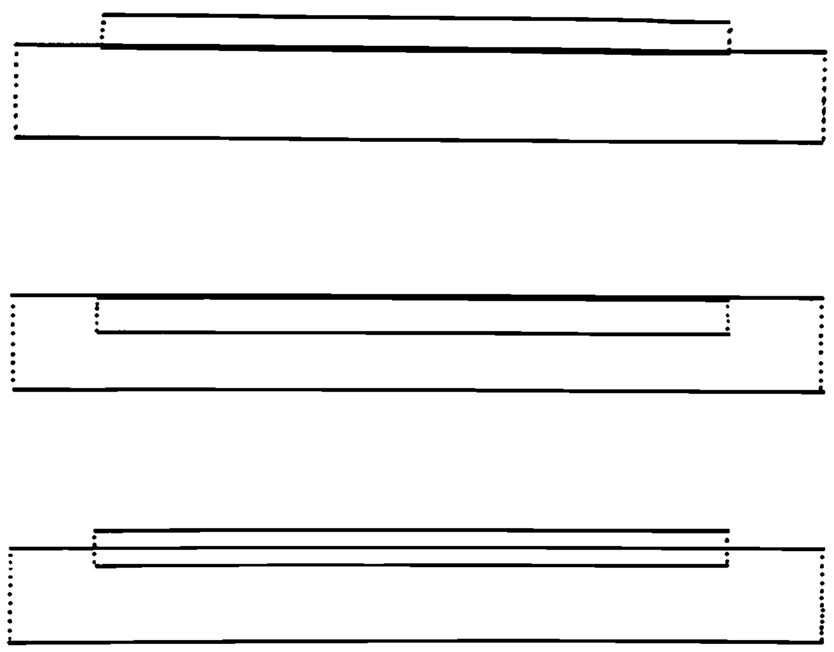


(i) A slot feature and (ii) the solids associated solids that occur in the above signed distance expression, C1.

Figure 5.9.4

Note that for a system to derive these constraints it has to know how a signed distance can be expressed in terms of DOF-variables. It has to know this or be able to derive this for every feature that occurs. Appendix 1 explains how this can be done.

It has not been proved that the constraints obtained by these methods are the best available. In fact, it is sometimes possible to tighten the constraints without reducing the class of possible real features that could be contained in the intersection of the zones. For example consider a planar feature with two zones associated with it. Figure 5.9.5(i) shows the extreme relative displacement of the zones. In this situation the only possible surface lies along the common boundary of the zones. However this surface could also arise when the zones have less displacement as shown in figure 5.9.5(ii). In fact all of the surfaces that could arise from the situation in figure 5.9.5(iii), where the zones overlap, could also arise in situation (ii). Therefore the maximum displacement of the zones could be taken as situation (ii) without reducing the class of possible real features.



(i)

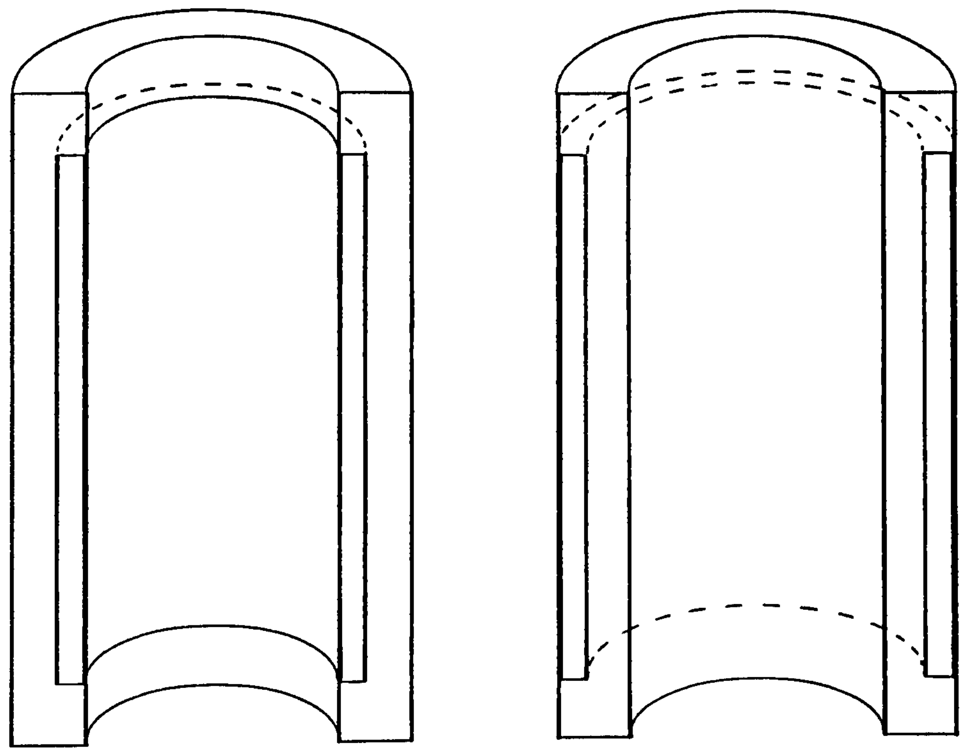
(ii)

(iii)

Two tolerance zones associated with a planar feature. All possible surfaces in situations (i) and (iii) could also be obtained in situation (ii).

Figure 5.9.5

A similar result is obtained for a feature of any shape if there is a form tolerance involved. For example, figure 5.9.6 shows the maximum and minimum size that need be taken by a cylindrical zone of form tolerance relative to a position tolerance zone. Compare this with figure 5.9.3.



The maximum and minimum size that need ever be taken by a form tolerance zone (lesser thickness) relative to a position tolerance zone (greater thickness) (cf. figure 5.9.3).

Figure 5.9.6

5.9.4. Datum-Defining Relationships

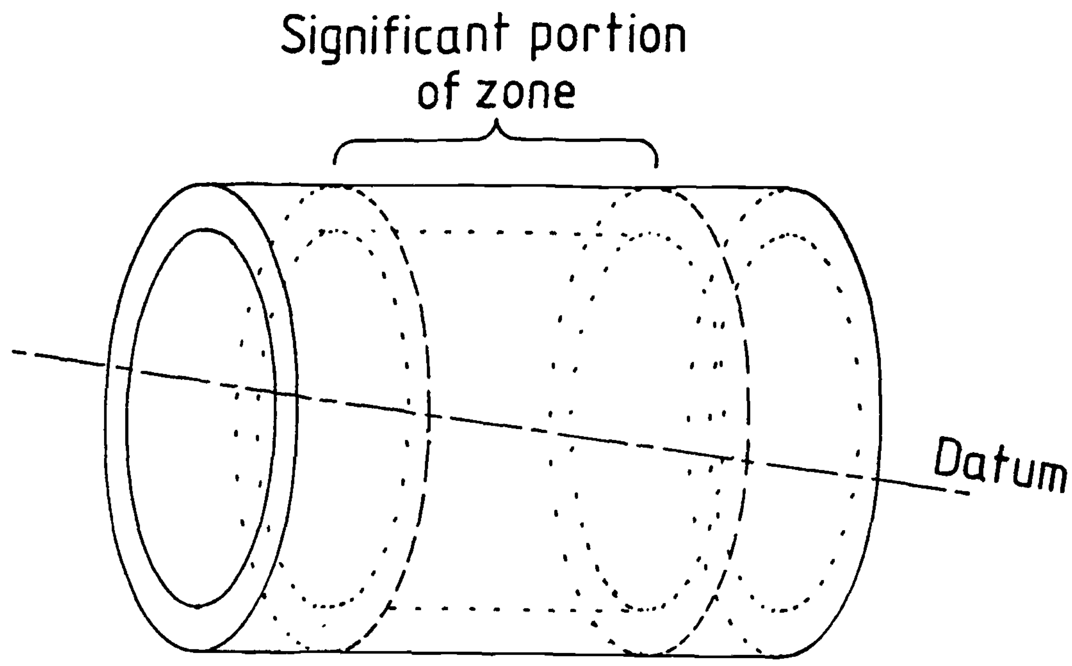
Every datum is defined by a planar feature or by a symmetric feature which may be simple or composite, though only simple features are considered in this section. In the example of section 5.1 datums C and D were defined by the large holes and so made relationships with their tolerance zones.

There are constraints on the relative position of a datum and the zones of its associated feature. For example, a line datum centred on a cylindrical feature is approximately parallel with the zone and is approximately coaxial with it in the significant portion of the zone as shown in figure 5.9.7. The maximum possible deviation of the datum from the axis of symmetry of the zone, inside the significant portion of the zone, depends on the possible deviation in the surface from its nominal shape and position. This in turn depends on the thickness of the zone.

The position of a datum relative to a symmetric feature of an actual part is constrained geometrically by the "measuring procedure" defined in section 4.2.5. Use is made of an expanded and orientated copy of the extended feature. It is called the "measuring solid" and can be expressed as $O(d;H')$ where H' is a copy of the extended feature. The number, d, in this expression will be called the <u>size</u> of the measuring solid.

Suppose that we require to find the constraints on the position of the datum relative to a zone $Z=O(T_1;H")-O(T_2;H")$. Also, suppose that a feature satisfies a size tolerance with parameters S_1 and S_2. (If Z is a zone of size tolerance then $S_1=T_1$ and $S_2=T_2$.) There are two geometric conditions that the measuring solid satisfies which will be referred to as Measuring Solid Constraints 1 and 2.

- **Measuring Solid Constraint 1.** The maximum required size of the measuring solid is that which is large enough to contain the significant portion of the size tolerance



The position of a datum is approximately colinear with a tolerance zone in the significant portion of the zone.

Figure 5.9.7

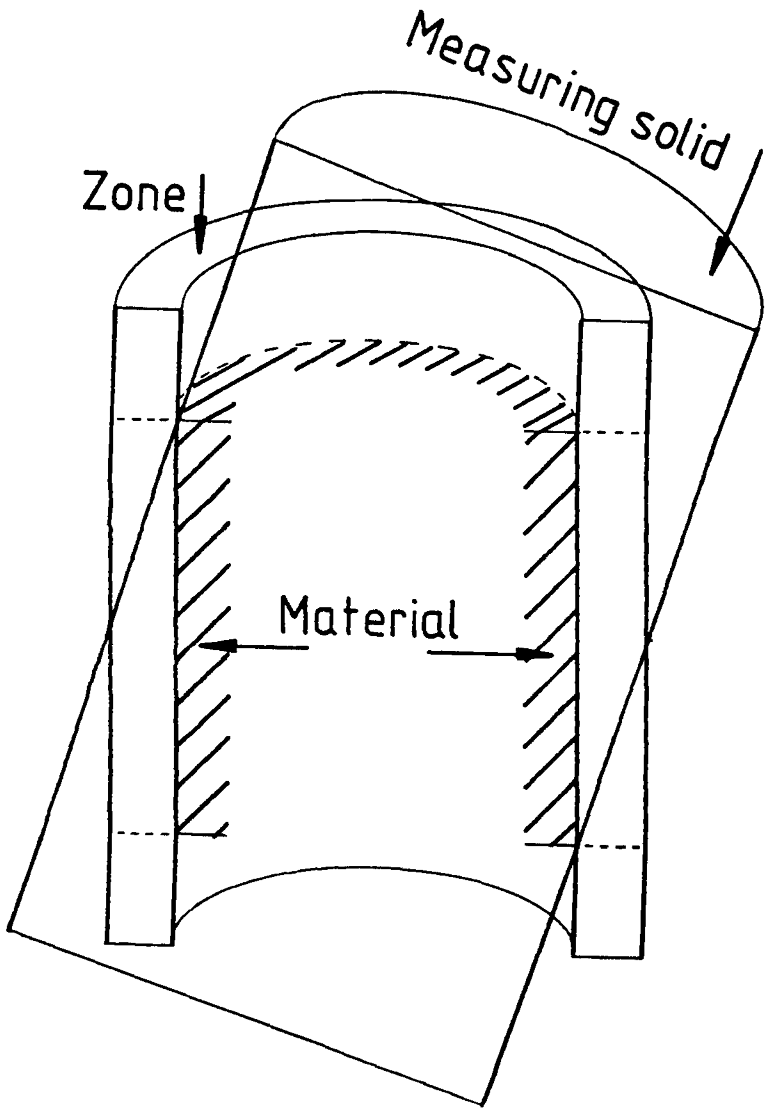
zone.

This ensures that it will be large enough to contain any possible real feature. If $d=S_1$, the larger of the size tolerance parameters, then the measuring solid can be orientated so that it contains the zone. Its surface will then be coincident with the surface of the size tolerance zone. It follows that d need never be greater than S_1.

- **Measuring Solid Constraint 2.** The measuring solid must enclose the material region of the zone Z.

This is because every instance of the part has material in the material region of the zone, at least close to the zone. The measuring solid must enclose the surface of the real feature or, equivalently, all material close to the feature.

Figure 5.9.8 shows a section through a cylindrical zone with the measuring solid at the most extreme inclination that satisfies the above constraints. The diameter of the measuring solid is (approximately) equal to the outer diameter of the zone and it contains the material region of the zone.



The maximum inclination of a measuring solid allowed by the two Measuring Solid Constraints in the case of a cylindrical feature.

Figure 5.9.8

The second constraint can be expressed in terms of signed distances as

$$\mathrm{sdist}(\mathrm{comp}(O(d;H')) , O(T_2;H")\cap E) \geq 0. \qquad C2$$

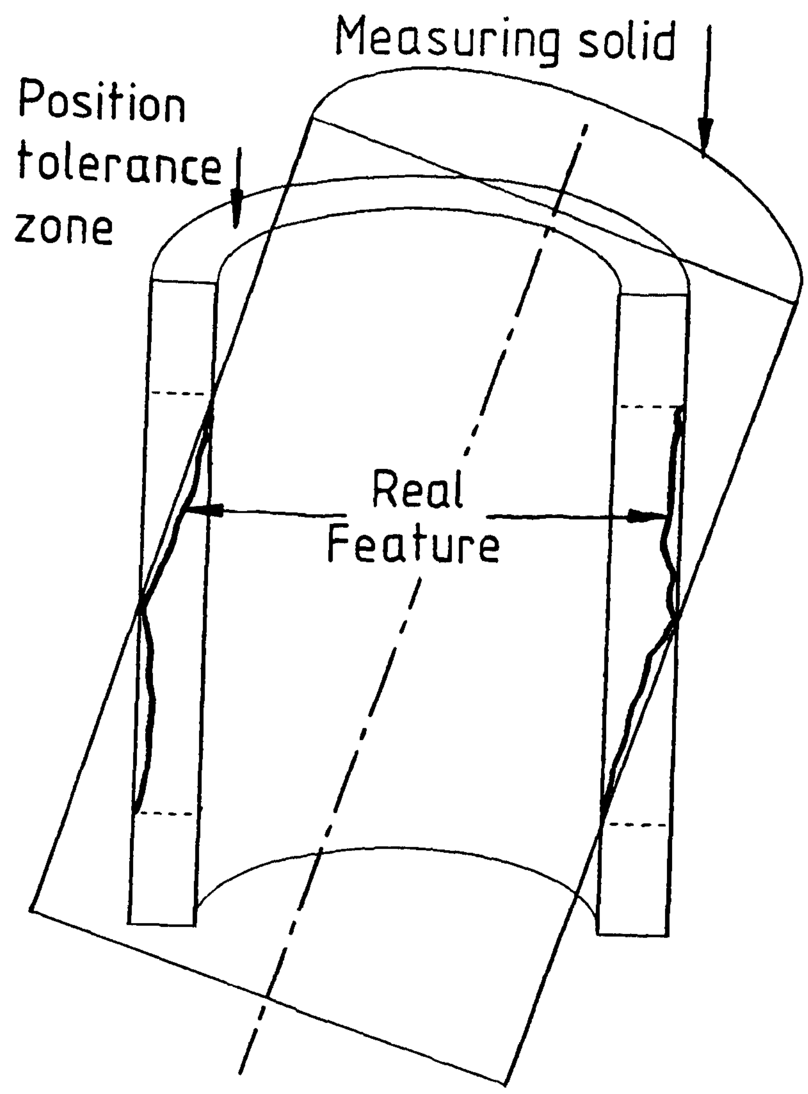
Since the constraints will be weakest when the measuring solid has maximum size, d can be replaced by S_1 in this expression to obtain an expression from which bounds on the relative position of H′ and H" can be obtained.

It is not obvious that these methods result in constraints which are the strongest attainable constraints. However, the following will show that for a cylindrical feature (and a slot feature), at least, the constraints obtained are realistic.

Figure 5.9.9 shows the same zone as figure 5.9.8 containing a real feature. The real feature has been chosen so that slightly more than half of its surface is inclined and the other half is parallel with the zone. A measuring solid enclosing this surface attains minimum size if it is parallel with the inclined part of the surface. (Note that, if less than half of the surface had been inclined then the measuring solid would attain minimum size by being parallel with the zone.) The inclination of the datum implied by this surface is the same as the maximum inclination derived from the geometric constraints above. A comparison of figures 5.9.8 and 5.9.9 will show this. Therefore, for a cylindrical feature, at least, the constraints derived from the geometric conditions are accurate.

As an example, suppose that C2 is applied to a cylindrical shaft feature with nominal length E defining a linear datum. The measuring solid is an infinitely long cylinder. The signed distance between the complement of the measuring solid, O(d;H′), and the material region of the zone $O(T_2;H")$ is given by,

$$d-T_2-\sqrt{(\delta y+(E/2)\delta\psi)^2+(\delta z+(E/2)\delta\phi)^2}.$$



The same zone as figure 5.9.8 showing that there does exist a surface which gives rise to this inclination of the measuring solid.

Figure 5.9.9

This expression must be greater than or equal to zero. d can be replaced by its upper bound, S_1, so that the constraints are as general as possible. Hence the constraint on the position of the datum relative to the tolerance zone is

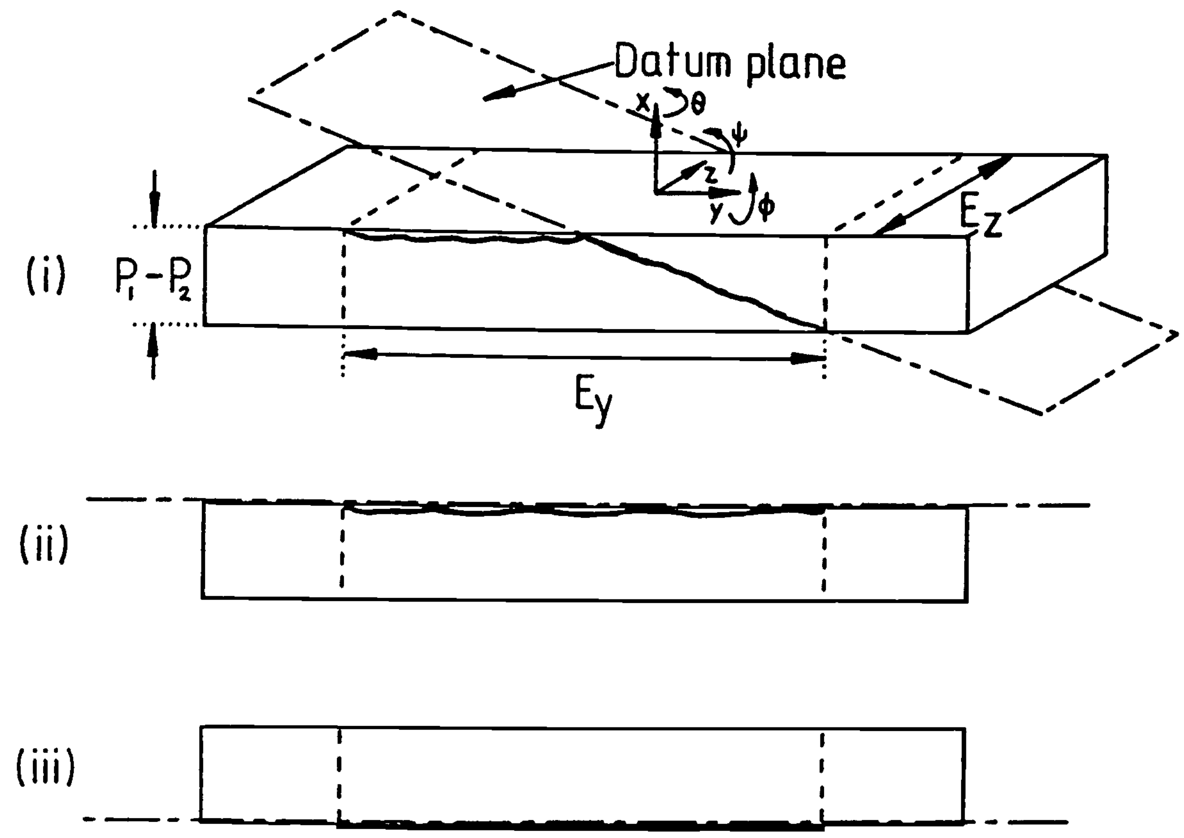
$$S_1-T_2-\sqrt{(\delta y+(E/2)\delta\psi)^2+(\delta z+(E/2)\delta\phi)^2} \geq 0.$$

This applies to a zone of any tolerance type. However, if the zone is of form tolerance then T_2 and T_1 are variable. Nonetheless, it can be assumed that $T_1 \leq S_1$ (see discussion associated with figure 5.9.6) and that $T_1-T_2=T_f$, the form tolerance parameter. Therefore, $T_2+T_f \leq S_1$ and so $T_f \leq S_1-T_2$. Hence S_1-T_2 can be replaced by T_f in the above.

Planar Features and other Simple Cases

So far the discussion of this section has dealt with datums defined using a measuring solid and so has not been applicable to planar features. For these it is necessary to use inspection to decide what are the worst possible inclinations and displacements of a datum relative to a tolerance zone. In a similar way constraints could be found for certain simply shaped commonly occurring symmetric features such as cylinders, cones and spheres. In this way constraints would be obtained more accurate than those obtained by the general technique. The constraints obtained in this way could be catalogued.

Examples of worst cases will be given for a planar feature with a position tolerance. Figure 5.9.10 shows sections through its zone and different possible shapes of the real feature. The surfaces were chosen to give datums with extreme inclinations and displacements. In figure 5.9.10(i) the surface was chosen so that the datum attains maximum inclination. Slightly more than half of the real surface is inclined and the rest is parallel to the zone. The best way to rest a plane against this feature is to let it be parallel with the inclined part of the surface. (If more than half of the surface had been inclined then the best location for the datum would be parallel with the zone.) In figures 5.9.10(ii) and (iii) the surface is such that the datum attains extreme translational displacement in either direction.



Extremes of possible inclination and translational displacement of a datum relative to a zone of a planar feature. They are found by considering possible surfaces that fit in the zone.

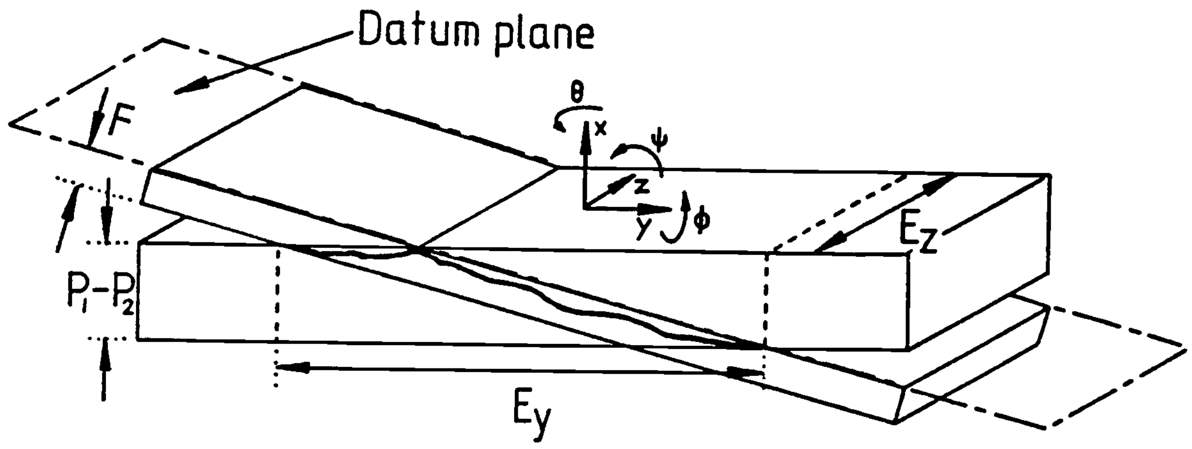
Figure 5.9.10

Suppose, that the nominal feature has a rectangular extent with dimensions E_y by E_z. The length of the inclined part of the surface in figure 5.9.10(i) is $E_y/2$ and the thickness of the zone is the difference between the tolerance parameters P_1 and P_2. This allows bounds on $\delta\psi$ to be determined and, similarly, bounds on δx and $\delta\phi$ can be found:

$$P_2 \le \delta x \le P_1,$$
$$-2(P_1-P_2)/E_y \le \delta\psi \le 2(P_1-P_2)/E_y,$$
$$-2(P_1-P_2)/E_z \le \delta\phi \le 2(P_1-P_2)/E_z.$$

Planar Features with more than one Tolerance Type

Suppose a rectangular planar feature has extent E_y by E_z and has a position tolerance with parameters, P_1 and P_2, and a form tolerance with parameter F. A datum defined by the feature is shown at its maximum attainable inclination in figure 5.9.11. By comparison with figure 5.9.10 it can be seen that the presence of the form tolerance prevents the datum from attaining such a large inclination.



A planar feature with two tolerances and the maximum inclination attainable by the datum.

Figure 5.9.11

Inspection of the geometry of the figure shows that the inclination of the datum is constrained by

$$-(P_1-P_2+F)/E_y \leq \delta\psi \leq (P_1-P_2+F)/E_y,$$

and similarly,

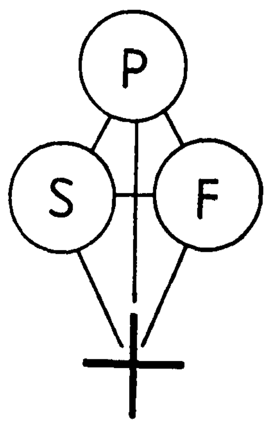
$$-(P_1-P_2+F)/E_z \leq \delta\phi \leq (P_1-P_2+F)/E_z.$$

Translation of the datum is constrained by

$$P_2 \leq \delta x \leq P_1.$$

Other Features with more than One Tolerance Type

If a feature has more than one tolerance type then the position of the datum is constrained relative to each of the zones. It would be convenient if the network of zones and datums for a feature with position, size and form tolerances defining a datum could be as shown in figure 5.9.12. The zones are constrained relative to one another by a relationship of the type described in section 5.9.3. Also, each zone makes a relationship with the datum. Constraints could be associated with each of these relationships as described earlier in this section using the two Measuring Solid Constraints.

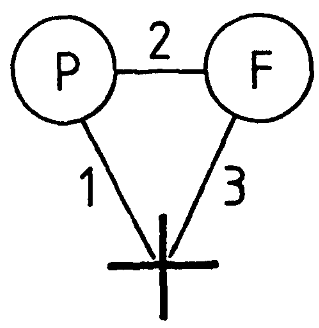


A datum makes a relationship with each zone of its associated feature. P=zone of position tolerance, S=zone of size tolerance, F=zone of form tolerance.

Figure 5.9.12

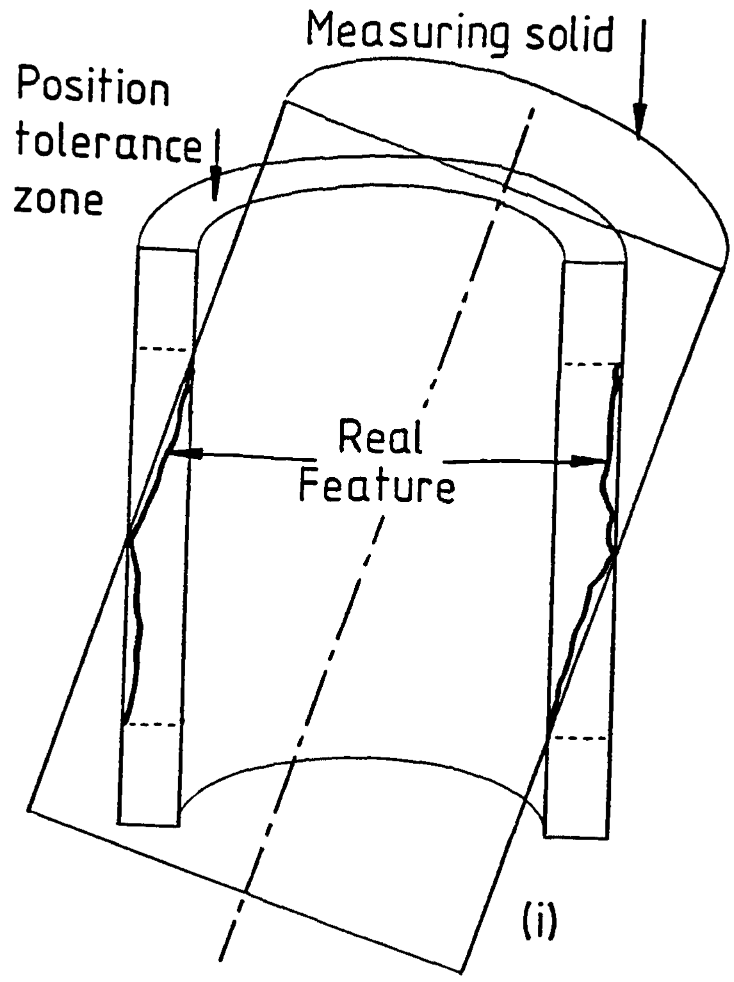
Unfortunately, the total constraints on the datum can be stronger than implied by constraints from individual arcs in this network. This is because the surface lies in the intersection of zones.

As an example, suppose a cylindrical feature has a form tolerance and a position tolerance and defines a datum. The network of relationships is the simplified version of figure 5.9.12 shown in figure 5.9.13. The relationships are numbered 1, 2 and 3. Figure 5.9.14(i) shows the maximum inclination of a datum assuming that the surface of the cylinder lies in its position tolerance zone. This is the bound on the inclination implied by the constraints of relationship 1 alone. Figure 5.9.14(ii) shows the maximum inclination assuming that the surface lies in its form tolerance zone. The form tolerance zone is at its maximum allowed inclination relative to the position tolerance zone. The resulting inclination of the datum is implied by relationships 2 and 3. Figure 5.9.14(iii) shows the actual maximum inclination of the datum when the real feature lies in the intersection of the zones.



The network of relationships for the example illustrated in figure 5.9.14. P=position tolerance, F=Form tolerance.

Figure 5.9.13



The maximum inclination of the datum when (i) the surface lies only in the position tolerance zone, (ii) the surface lies only in the form tolerance zone and (iii) the surface lies in the intersection of the zones. In (iii) the measuring solid is at a smaller inclination than in either of (i) or (ii).

Figure 5.9.14 (Continued on next page)

Hence, the total constraints on the position of the datum obtained from the network in figure 5.9.13 are not maximally strong. However, it can be guaranteed that they are weaker than reality because the constraints in each arc are correct. Therefore, the constraints obtained are sufficient in the absence of better techniques.

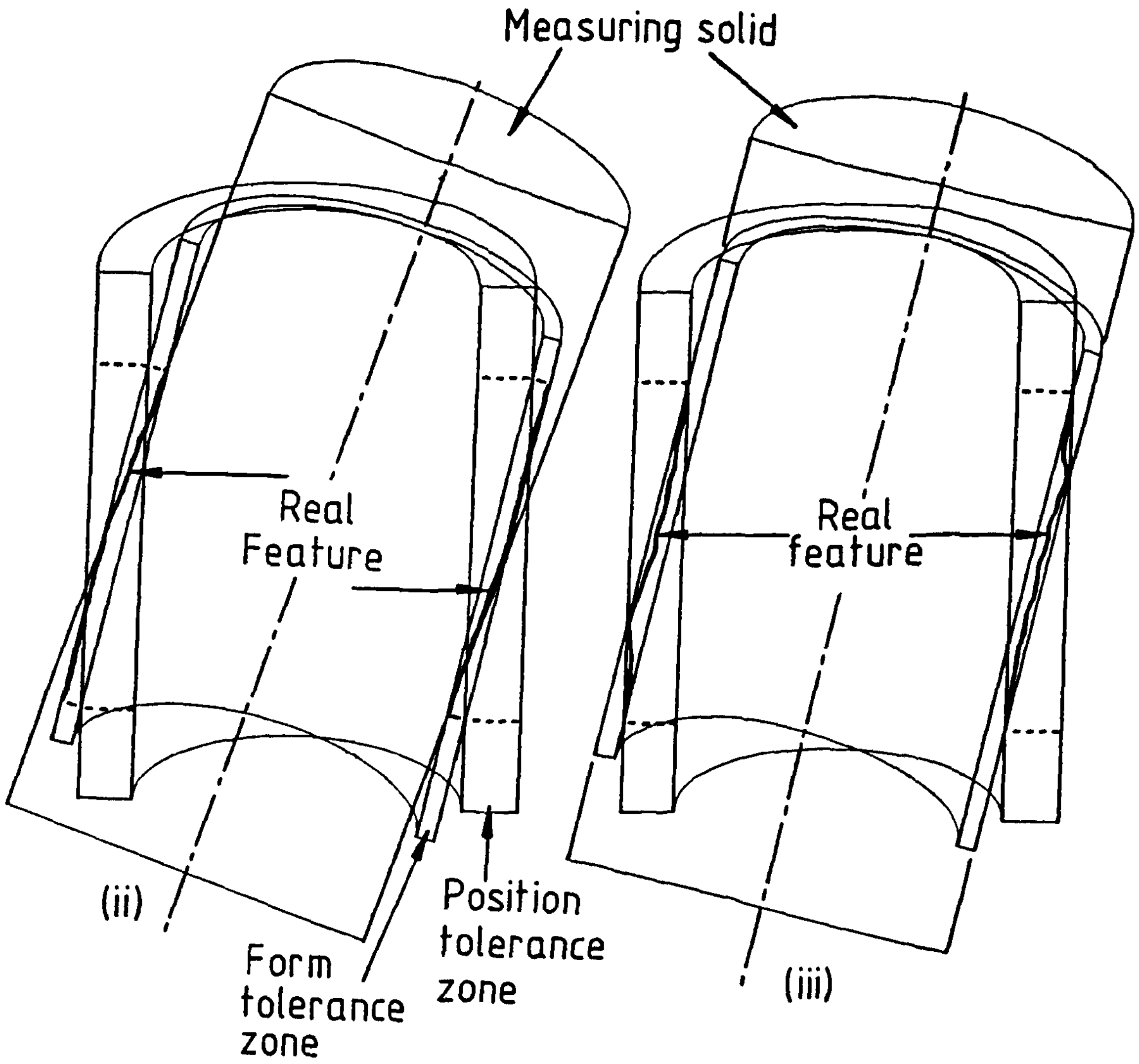


Figure 5.9.14 continued

Secondary Datums

The position of a secondary datum is constrained relative to the tolerance zones of its associated feature but the constraints are weaker than they would be if the datum were primary. This is obvious from figure 5.1.3 where the primary datum was forced to be parallel with the overall trend of its associated feature but the secondary datum only touched its associated feature at one point.

In section 5.9.3 the constraints between a secondary datum and a primary datum were discussed. Some degrees of freedom of the secondary datum are determined by its relationship with the primary datum and as a result are not affected by the feature associated with the secondary datum. Therefore, there are certain degrees of freedom which do not occur in the inequalities of the relationships between the secondary datum and its associated tolerance zones.

It is possible to derive the constraints that a secondary datum has with a zone from the constraints that the datum would have if it were primary. Certain variables have to be eliminated.

Let C_1 and C_2 be the sets of constraints that primary and secondary datums, respectively, would have with a zone and let D_1 and D_2 be the variables appearing in C_1 and C_2. The solution set of C_2 should be the projection of the solution set of C_1 from space(D_1) to space(D_2).

Firstly, define <u>space(V)</u> to be the cartesian-product of the sets of values (usually all of real numbers) which can be taken by variables in V. Then, C_2 satisfies the following: $\mathbf{p} \in$space(D_2) is a solution of C_2 if and only if there is a $\mathbf{q} \in$space(D_1) which is a solution to C_1 such that the d-components of **p** and **q** are equal for all δd in D_2.

The inequalities in C_2 can be derived from those in C_1 as follows. Let DOF-variable, δd, be a member of D_2. Find the SUP and INF of δd in terms of variables in $D_2-\{\delta d\}$. Then two inequalities can be constructed with δd less than or equal to its SUP and with δd greater than or equal to its INF. C_2 consists of all such inequalities with δd in D_2. Formally,

$$C_2 = \left\{ \text{INF}(\delta d, C_1, D_2-\{\delta d\}) \leq \delta d \leq \text{SUP}(\delta d, C_1, D_2-\{\delta d\}) \ : \ \delta d \in D_2 \right\}$$

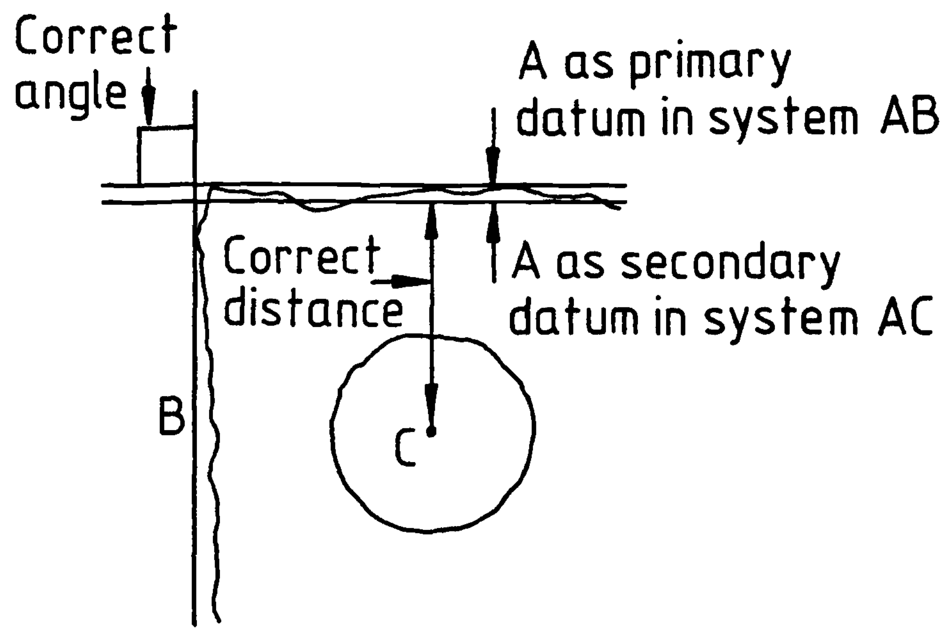
It is possible for a datum to appear in more than one datum-system and for it to be primary in one datum-system and secondary in another (for example). In the example of section 5.1 datums A and B formed a datum-system in which datum A was primary. Also datums A and C formed a datum-system in which A was secondary. In the first case A is constrained **only** by its associated feature but in the second case it is also constrained to be the correct distance from C. However, since the constraints on A in the two situations are different it follows that the two occurrences of A should be represented as separate entities. Similarly, A appears in datum-system AD. Therefore, there should be three distinct nodes in the zone-datum network corresponding to A.

This is quite reasonable if one thinks of the manufacturing process. The two datum systems represent two different ways in which the part was clamped while the holes were drilled. Since the face associated with datum A is imperfect there is no reason to suppose that the part will rest in identical positions in the two set-ups.

5.9.5. **Approximations in Constraints**

The inequality constraints obtained in the last two types of relationship described above contain approximations. Approximations may be acceptable for two reasons. Either they are small enough to be ignored or they result in an overestimate of the class of zone datum structures (and hence an overestimate of the variational class) as described in section 5.3. Thus it can be acceptable to slacken constraints. Situations where approximations occur are the following.

Firstly, the extent of a real feature is approximated to be the same as the nominal feature. Wherever a value representing the extent of a feature appears in one of the inequalities it should actually be a variable bounded according to the variation



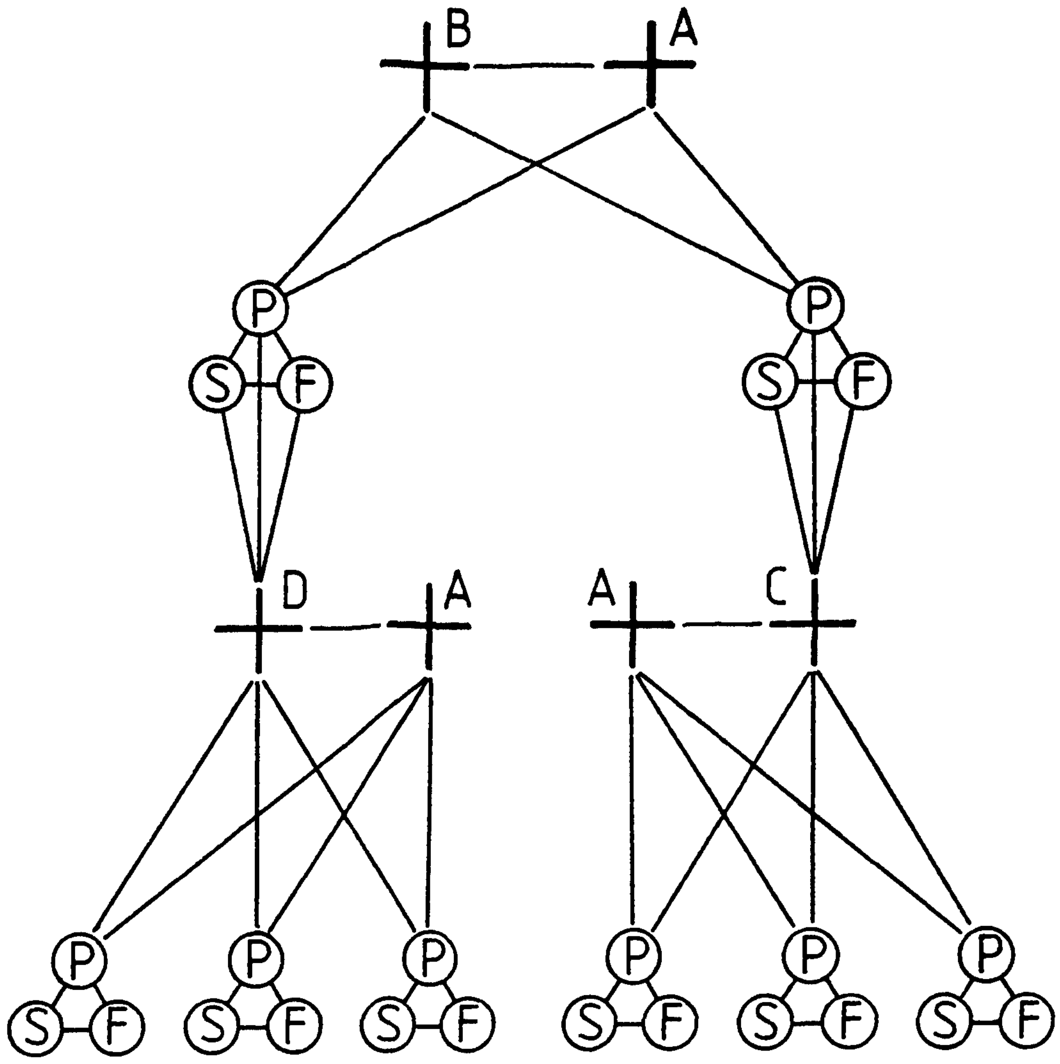
Two instances of datum A one of which is a primary datum in system AB and the other is a secondary datum in system AC.

Figure 5.9.15

in size of the feature. However, this type of approximation can be ignored when tolerances are much smaller than the extent of features.

Secondly, it may be necessary to approximate the extent of a feature if its extent has a complex shape. This is because the resulting inequalities could be too complex to be handled by the system. The extent can be underestimated and simplified so that the resulting constraints on the datum are slackened and simplified.

Thirdly, the techniques for finding constraints between zones associated with the same feature and between a datum and the zones of its associated feature can produce constraints that are weaker than would occur in reality. This is because the geometric constraints from which the algebraic constraints are derived do not present a complete picture. Again these are acceptable because the constraints are slackened and the variational class is



The complete zone-datum network for the part in figure 5.1.1.

Figure 5.9.16

overestimated.

5.9.6. The Example of Section 5.1 Again

Figure 5.9.16 shows the complete formalised network for the part presented in section 5.1 with the addition of a size and a form tolerance applied to each feature. Each feature is represented by three nodes to represent the three tolerance zones of position, size and form associated with it. The following is a list of situations where relationships occur.

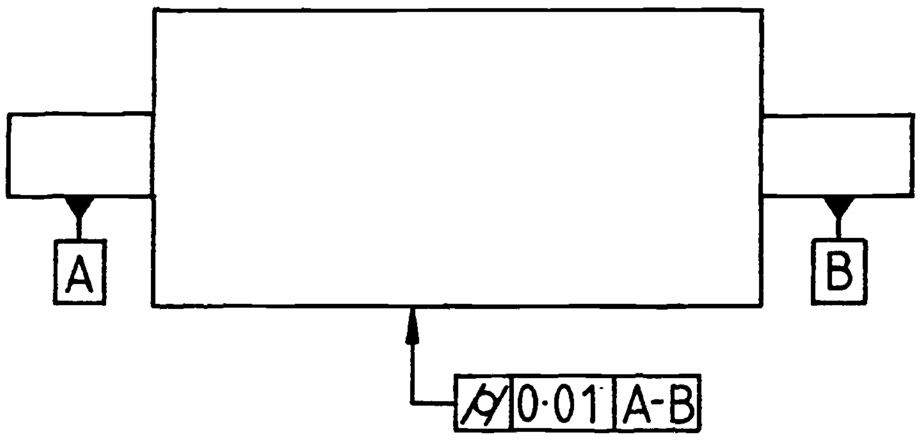
1. Between a position-tolerance zone and the two datums that locate it;
2. Between the two datums of each datum-system;
3. Between tolerance zones associated with the same feature;
4. Between a datum and the zones of the feature that defines it.

5.10. COMPOSITE FEATURES

A composite feature is a set of simple features. There are two reasons for defining composite features.

Firstly, a datum may be defined using more than one simple feature. For example, the ends of the shaft shown in figure 5.10.1 could together define a line datum denoted as A-B in standard practice. Requicha's formalism requires that a composite feature comprising the simple features (A and B in the figure) be defined and that the datum be associated with this composite feature.

Secondly, a tolerance may be applied over a larger portion of the surface of a part than a single simple feature. Again Requicha's formalism requires that a composite feature be defined as the relevant surface portion and that the tolerance be applied to the composite feature. A composite feature may exist for both of these reasons or for just one of them.



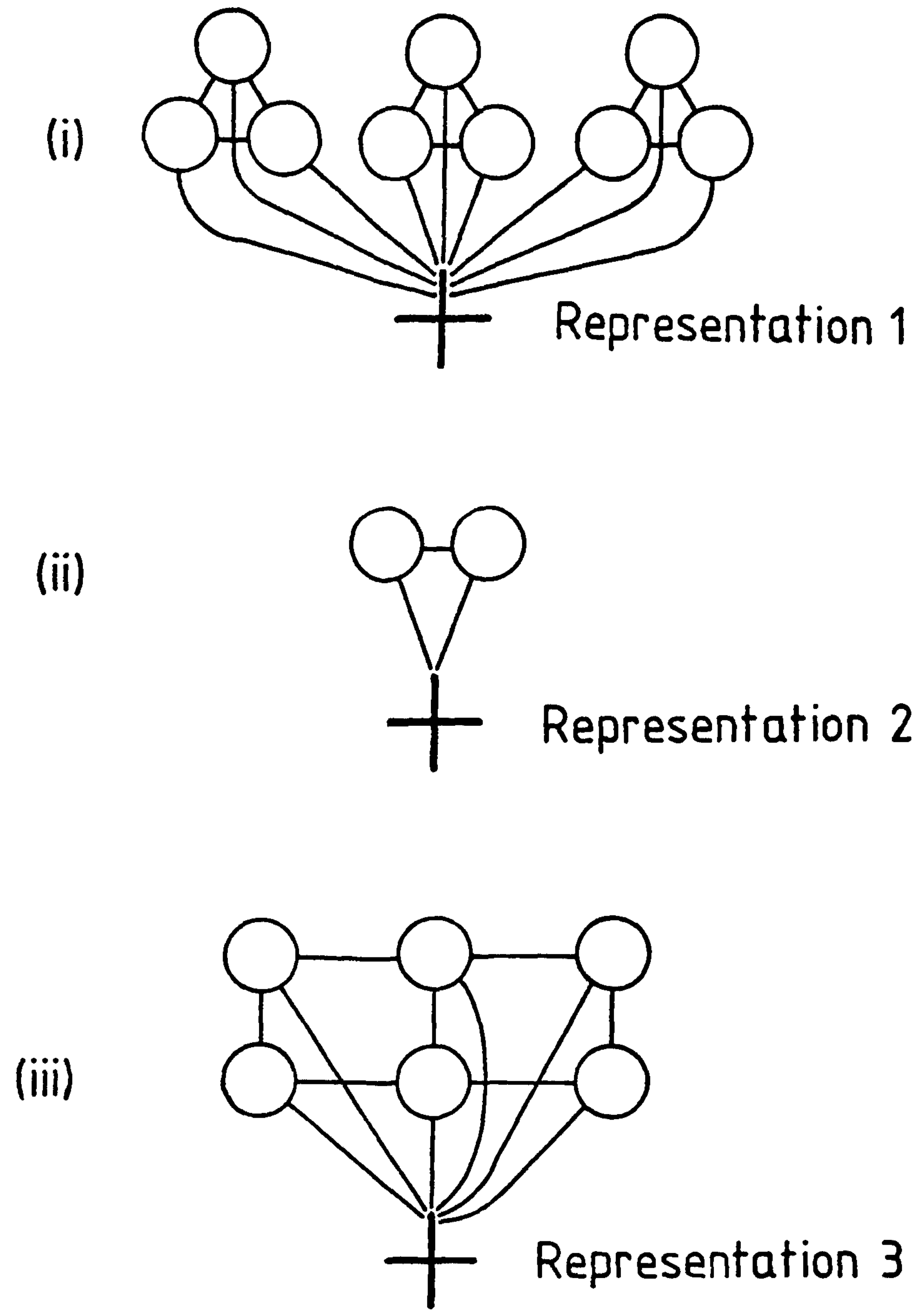
The two ends of the shaft define a datum denoted **A-B**.

Figure 5.10.1

Three different ways will be presented for representing composite features each of which is compatible with the zone-datum network described in previous sections. The first representation is used when the composite feature defines a datum but the simple features have been toleranced separately. The second and third representations are applicable when a tolerance has been applied to a whole composite feature.

Representation 1: Suppose that each simple feature comprising the composite feature has been toleranced separately and that the composite feature defines a datum. The composite feature can be considered as individual simple features when representing the zones as nodes in the network. The position of the datum is affected by the surface of each simple feature and is therefore constrained relative to each zone of each simple feature. As an example, the portion of zone-datum network for a composite feature defining a datum and consisting of three simple features each with three tolerances is shown in figure 5.10.2(i).

Representation 2: Suppose that tolerances have been applied to the whole of the composite feature. There is one zone for each tolerance and each should contain the real feature. Each zone could appear as a single node in the network resulting in a zone-



The zone-datum networks of the three representations of composite features. These can be generalised to cases with different numbers of simple features or tolerance types.

(i). Each simple feature toleranced separately. Three simple features each with three tolerance types.

(ii). A composite feature toleranced as a whole with nodes representing complete zones. Two tolerance types.

(iii). A composite feature toleranced as a whole with each node representing a subzone. Three simple features each with two tolerance types.

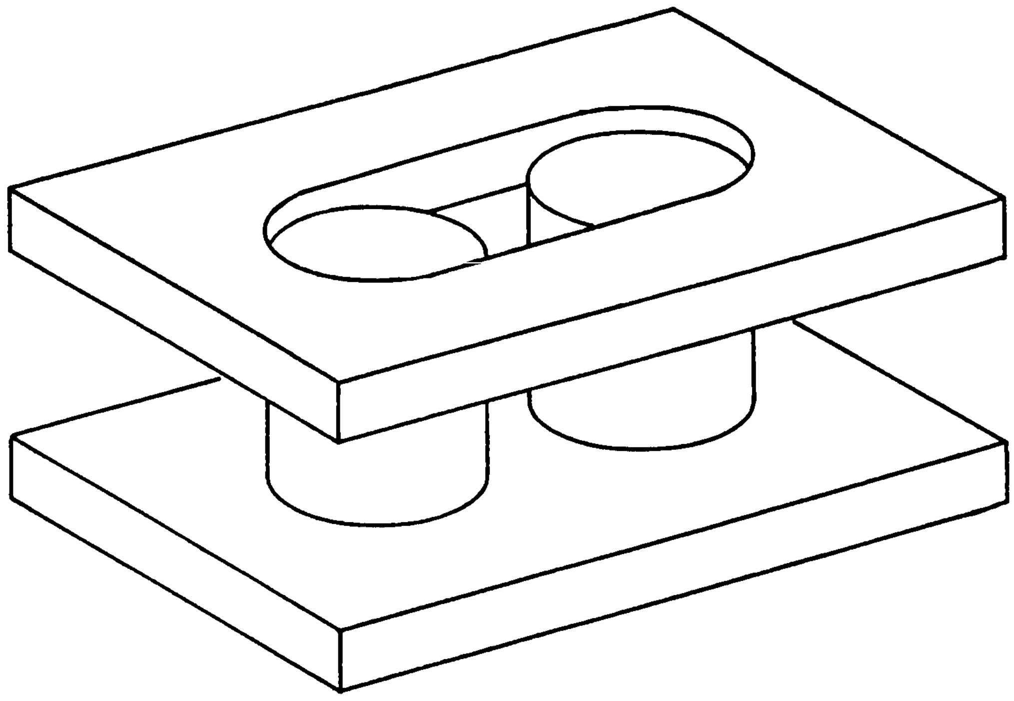
Figure 5.10.2

datum network which is identical to that arising from a simple feature. For example, a composite feature with two tolerances defining a datum would have the zone-datum network shown in figure 5.10.2(ii).

Representation 3: This representation has been developed because it is often useful for the simple features that make up a composite feature to appear explicitly in the network. This is done by splitting the zones into subzones (in a way to be described later) each corresponding to a simple feature. Each subzone appears as a node in the network. Figure 5.10.2(iii) shows the resulting portion of zone-datum network for a composite feature assuming there are two tolerances and three simple features. Also included is a datum defined by the composite feature. Horizontal arcs in this diagram link the subzones that comprise a single zone and these will be described later. The vertical arcs represent relationships expressing the fact that there must be room in the intersection of the subzones for a real feature. The datum makes a relationship with each subzone.

Representation 3 is important when a part is to be mated with others with "incompatible" feature allocations (see discussion in section 4.5) or when a composite feature is to be mated with a set of simple features which have not been defined as a composite feature. For example, it is quite feasible that there could be a composite feature consisting of a round ended slot into which two parallel cylindrical pegs are inserted (figure 5.10.3). Constraints between two contacting features can be most easily represented if the individual simple features appear explicitly in the zone-datum network. This point will be returned to in chapter 6.

The reader should be aware that the first representation is separated from the second and third by the way the part has been toleranced by a designer. The second and third representations are chosen according to which is most convenient, the second being the simplest and the third being the most flexible.



Two simple features (shafts) inserted into a composite feature (round-ended slot). This shows how feature allocations of two mated parts may be incompatible.

Figure 5.10.3

The rest of this section explains how constraints can be associated with the arcs in each of the three representations though many of the relationships in the three representations can be handled in the same way as the relationships that occur with simple features.

In representation 2 all the relationships can be handled in the same way as those discussed in the last section. There are (1) relationships between zones and (2) relationships between a zone and the datum. These types of relationship were the subject of sections 5.9.3 and 5.9.4. The only difficulty is that there is a much greater variety of types of composite features than there is of simple features. A signed distance expression has to be known for each shape of feature that will occur. An algorithm that could do this is outlined in Appendix 1

Other relationships which need no further explanation are those between zones in representation 1. This is because this type of relationship (between zones of a simple feature) was described in section 5.9.3.

The rest of this section is divided into two parts.

- The first shows how a zone of a composite feature is split into subzones and how constraints can be associated with the all of the arcs in representation 3. It turns out that the constraints are quite straightforward.
- The second part shows how constraints are associated with the relationships in representation 1 between a datum and zones of individual simple features.

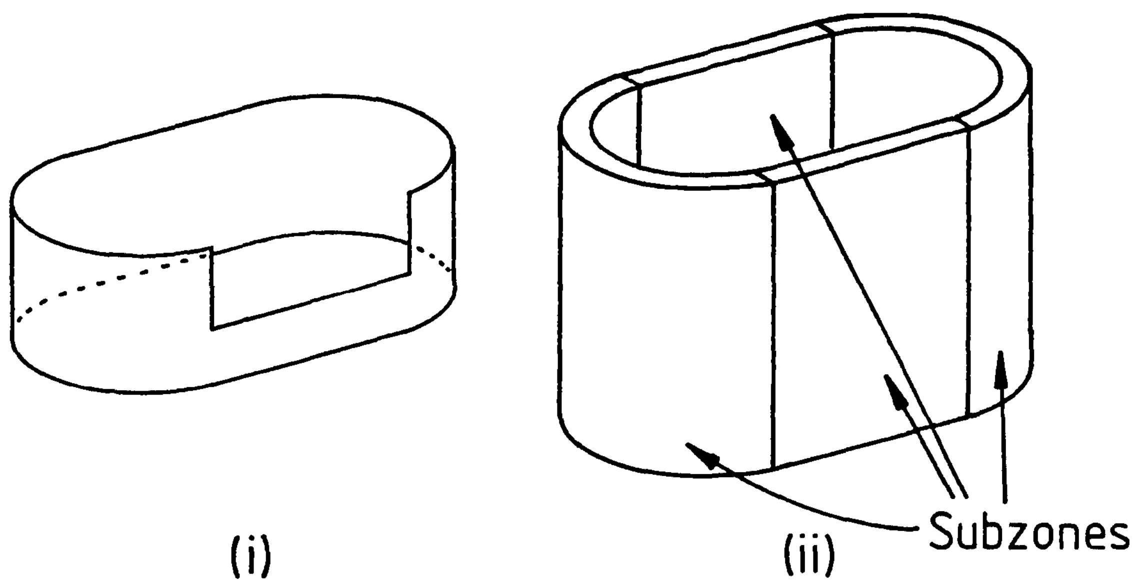
5.10.1. Splitting the Zone of a Composite Feature into Subzones

The general idea is to consider the zone of a composite feature as a union of subzones each of which corresponds to one of the simple features. Each subzone is given a corresponding node in the network. If the subzones together are to constrain the real feature in the same way as the entire zone then the union of the subzones should equal the original zone. Therefore, the subzones should satisfy the following constraints.

1. Their relative positions must be the same as the relative positions of the associated nominal simple features.
2. The offsets of the offset solids used to generate the subzones must all be identical.

It is convenient if the subzones are disjoint from one another.

As an example, consider the round ended slot which is shown along with its tolerance zone in figures 5.10.4(i) and (ii). There are four simple features, two planar and two cylindrical. The tolerance zone is shown divided into four subzones each corresponding to one of the cylinders or one of the planes. The zone (and hence the subzones) extends infinitely upwards and downwards.



A composite feature and its tolerance zone divided into four subzones. The zone actually has infinite extent upwards and downwards.

Figure 5.10.4

Suppose that the slot has both a size tolerance and a position tolerance applied to it. Then, there are two zones each divided into four subzones. There are four situations where relationships occur in this network and these will be dealt with in turn.

Firstly, each subzone of position tolerance is correctly positioned relative to a datum. These relationships are of the type discussed in section 5.9.1.

Secondly, the horizontal arcs between subzones have constraints which say that they have the same position relative to one another as the nominal simple features. Each of these relationships fixes some of the six DOF-variables to zero (cf. the relationship types described in sections 5.9.1 and 5.9.2).

Thirdly, the vertical arcs express the fact that the intersection of subzones of the same simple feature must be large enough to contain the real feature. The constraints can be evaluated in the same way as constraints between normal tolerance zones (as opposed to subzones). For example, if a separate tolerance were applied to each of the simple features representing the cylindrical ends of the slot in figure 5.10.4 then the associated tolerance zones would be complete cylinders. This contrasts with the subzones which are half cylinders. However, the significant portions of these zones and subzones are identical. Since the constraints between zones of the same feature are determined by the shape of their significant portions it follows that the methods described in section 5.9.3 can be used to find constraints between subzones.

Fourthly, the datum defined by the feature makes a relationship with each of the subzones. However, it is more convenient to consider the constraints that this datum has with each complete zone. This requires using the two Measuring Procedure Constraints from section 5.9.4 with the measuring solid and the zone derived from the entire feature. It is necessary to know the signed distance between two solids derived from the composite feature. The constraints can then be applied to each of the arcs between the datum defined by the feature and a subzone. This results in degeneracy since each the arcs leading to an associated set of subzones obtain the same constraints. However, the degeneracy will not cause any problems.

In conclusion, the tolerance zone of a composite feature can be represented as a network with nodes representing subzones which correspond to simple features. This method is compatible with the zone-datum network built up over the previous sections. The constraints associated with the arcs are derived in ways which are similar to those for simple features. Therefore, only a short discussion about these constraints was required here.

The existence of simple features is explicit in the network of this representation and so allows mated parts with unmatching feature allocations to be handled.

5.10.2. Datums Defined by Composite Features with Features Toleranced Individually

This section explains the derivation of constraints on the position of a datum in representation 1 (figure 5.10.2(i)). Each simple feature is toleranced separately but the datum is defined by the entire composite feature. The situation is more complex than previous examples of constraints on datums. This is because the position of the datum is affected by the surface of each simple feature and these are entirely independent. Each is contained in different zones whose relative positions are not directly known. In the most general case each zone would be located relative to a different datum. The datums could be constrained relative to one another in a complex way. This contrasts with the simpler situation described above where the subzones containing portions of a feature all had well defined relative positions.

Note first that the measuring solid of a composite feature may consist of disjoint components. For example, consider a feature consisting of two parallel holes. However, the components have fixed positions relative to one another and each component is subject to the same expansion or contraction. Thus if an extent solid H can be split into two disconnected components H_1 and H_2 then the measuring solid can be expressed as $O(d;H_1') \cup O(d;H_2')$ where H_1' and H_2' are obtained from H_1 and H_2 by applying the **same** rigid transformation.

The two Measuring Solid Constraints given for simple features in section 5.9.4, are used here adapted slightly to be relevant to composite features.

- Firstly, the maximum required size of the measuring solid is such that it can contain all of the significant portions of the size tolerance zones of the simple features that make up the composite feature.

The relative positions of the size tolerance zones are not known, directly. In the simple feature case the size of the zone was the maximum required size of the measuring solid but here the size depends in a complex way on the constraints on the relative positions of the zones.

- Secondly, the measuring solid must contain the material region of each zone.

There are two stages to the analysis corresponding to these two constraints. First, the maximum required size of the measuring solid must be determined. Next, the constraints on the position of the measuring solid with this size must be determined relative to the material region of each zone. These constraints can be applied to each of the arcs between the datum and the zones.

STAGE 1: Determining the Maximum Size of the Measuring Solid

This was a trivial operation in the case of simple features and with composite features toleranced as a whole but here the situation is complicated by the variable positions of the individual zones.

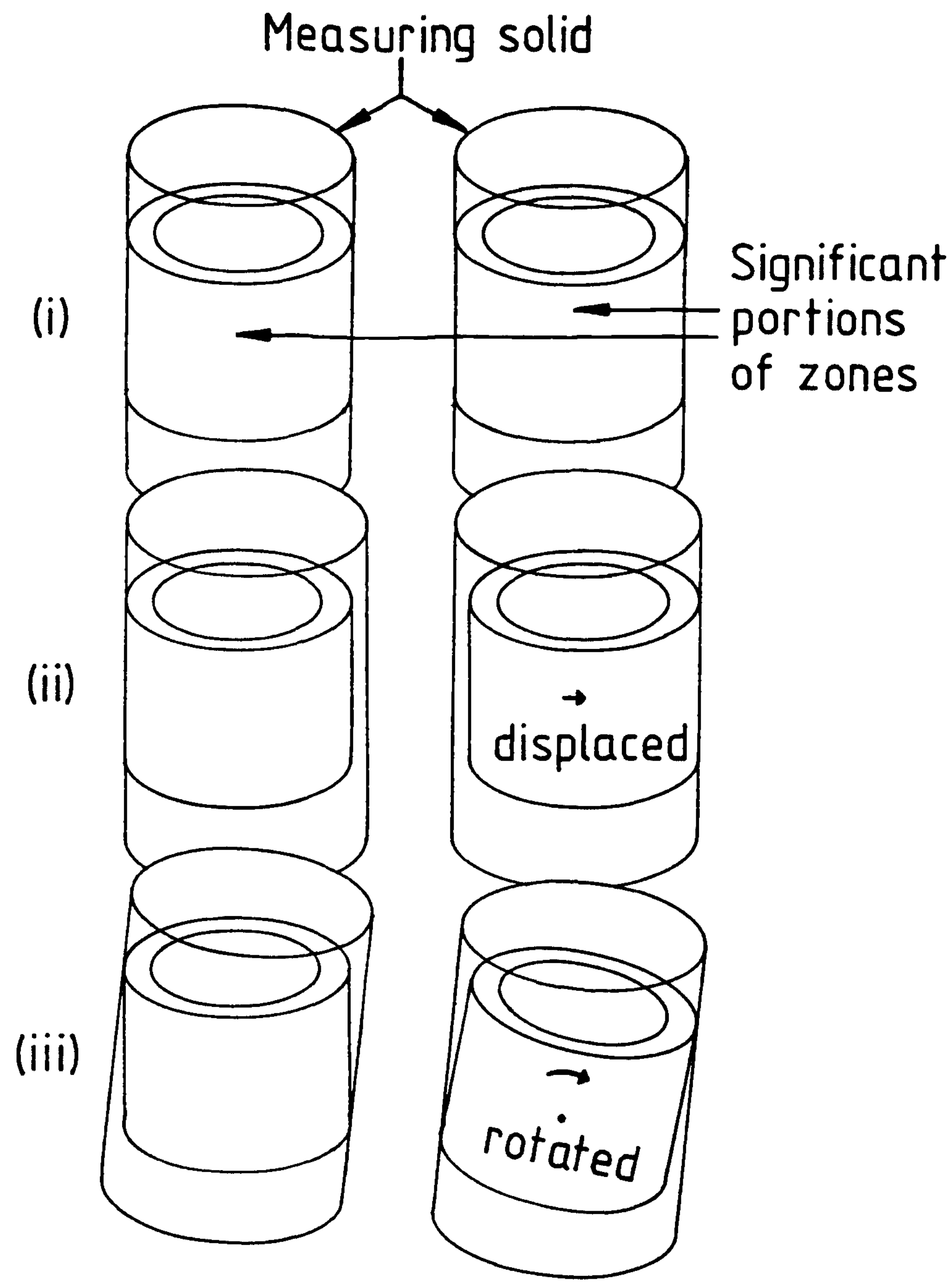
Remember that the maximum required size of the measuring solid depends only on size tolerances. Therefore, in STAGE 1 all references to tolerance zones refer to size tolerance zones.

As an example consider a composite feature consisting of two parallel cylindrical shafts. Its measuring solid consists of two infinitely long parallel cylinders with variable radii but with the distance between their axes fixed equal to the distance between the axes of the nominal features. In figure 5.10.5(i) the size tolerance zones of the simple features are shown at their nominal positions. The maximum size of the measuring solid that is required for a real feature lying inside these zones is such that its surface is coincident with the "outer" surface of the zones (ie. the surface next to the air region of the zone). However, if the zones are moved away from their nominal positions then the size of the measuring solid must be increased.

For instance, suppose that one of the zones is displaced translationally as in figure 5.10.5(ii). It is not possible to enclose the significant portions of the zones simply by translating the measuring solid. (Recall that the distance and angle between its two components is fixed.) Instead it is necessary to increase the size of the measuring solid. The increase in size is half of the displacement undergone by the zone.

Also consider that one of the cylinders is rotated (figure 5.10.5(iii)). Again the measuring solid must be increased in size if it is to enclose the significant portions of the zones. The amount of increase is half the change in angle multiplied by the length of the nominal feature.

The maximum size of the measuring solid, therefore, depends on how the positions of the zones are constrained: the further they can deviate from their nominal positions the larger is the maximum required size of the measuring solid. A method will be given for finding the maximum size of the measuring solid assuming



The maximum required size of a measuring solid of a composite feature with zones with variable position. Only the significant portions of the zones are shown. The measuring solid consists of two disjoint parallel cylinders whose axes are separated by a fixed distance. The measuring solid is shown at its maximum required size for the zones in the positions shown. In (i) the zones are at their nominal positions. In (ii) one zone has been displaced to the right. In (iii) one zone has been rotated clockwise.

Figure 5.10.5

that these constraints are known. In general the constraints are the result of combining constraints attached to the arcs in the zone-datum network. The next section will explain how this done.

There are three stages to finding the maximum size of the measuring solid.

- Stage 1: Assume that the zones (of size tolerance) and the measuring solid all have arbitrary positions. Find an expression for the smallest size of the measuring solid that will enclose the significant portions of the zones in terms of the positions of all these items.
- Stage 2: Now fix the position of the measuring solid but keep the positions of the zones arbitrary. The position of the measuring solid is selected so that its size can be decreased as much as possible while containing the zones. Find an expression for the size of the measuring solid under these conditions in terms of the positions of the zones. This involves taking the INF of the expression derived in Stage 1 over appropriate variables and constraints.
- Stage 3: Let the positions of the zones be constrained by some set of constraints. The position of the measuring solid is again constrained as in stage 2. Find the maximum size of the measuring solid under these conditions. This involves taking the SUP of the expression resulting from stage 2 over the constraints on the positions of the zones.

To explain how these calculations can be carried out some notation must be introduced to represent the positions of the zones and measuring solid. Let the volumes of space occupied by the significant portions of the size tolerance zones be denoted

$$S_1(\mathbf{p}_1),\ S_2(\mathbf{p}_2),\ \ldots,\ S_n(\mathbf{p}_n),$$

where $\mathbf{p}_1$, $\mathbf{p}_2$, ..., $\mathbf{p}_n$ are the positions of the zones as vectors of DOF-variables. Nominal positions are represented by these being zero. The values of $\mathbf{p}_i$, $i=1,\ldots,n$ are constrained by some set of constraints which will be denoted, Cz (Constraints on the zones).

The region of space occupied by the measuring solid (MS) can be written as the union of n components corresponding to the simple features:

$$MS(\mathbf{p},s) \;=\; MS_1(\mathbf{p},s) \cup MS_2(\mathbf{p},s) \cup \ldots \cup MS_n(\mathbf{p},s)$$

where $\mathbf{p}$ is the position of each component relative to its nominal position as a vector of DOF-variables and s is the offset used to produce the measuring solid from the copy of the extended feature and is referred to as the "size" of the measuring solid. Note that, the position and size of each component of the measuring solid are the same.

Stage 1 involves finding the minimum value of d for which the measuring solid, $MS(\mathbf{p},d)$ contains each of the significant portions of the zones. It turns out that this is the negative of the minimum of signed distances between $comp(MS_i(\mathbf{p},0))$ and $S_i(\mathbf{p}_i)$ for $i=1,\ldots,n$. This is proved by making a series of inferences, as follows, where "<=>" should be read "if and only if".

d is the minimum value for which $MS(\mathbf{p},d)$ contains all of $S_1(\mathbf{p}_1)$, ..., $S_n(\mathbf{p}_n)$.

<=> d is the minimum value for which $comp(MS(\mathbf{p},d))$ does not intersect any of $S_1(\mathbf{p}_1)$, ..., $S_n(\mathbf{p}_n)$.

<=> d is the minimum value for which $sdist(\; comp(MS_i(\mathbf{p},d)) \;,\; S_i(\mathbf{p}_i) \;) \geq 0$ for $i=1,\ldots,n$.

<=> $sdist(\; comp(MS_i(\mathbf{p},d)) \;,\; S_i(\mathbf{p}_i) \;) = 0$ for some i between 1 and n and this expression is less than or equal to zero for all other i between 1 and n.

[Because decreasing d decreases the signed distance expression.]

$$\Leftrightarrow \min_{i=1,..,n}[\ \mathrm{sdist}(\ \mathrm{comp}(MS_i(\mathbf{p},d))\ ,\ S_i(\mathbf{p}_i)\ )\ ] = 0$$

$$\Leftrightarrow \min_{i=1,..,n}[\ \mathrm{sdist}(\ \mathrm{comp}(MS_i(\mathbf{p},0))\ ,\ S_i(\mathbf{p}_i)\ )\ ] = -d.$$

[The last implication derives from the fact that $MS_i(\mathbf{p},d)$ is an offset solid, with offset d, of some solid and that

sdist(A,B) = sdist(O(d;A),B)-d

and hence

sdist(comp(A),B) = sdist(comp(O(d;A)),B)+d.]

It follows that the result of stage 1 is

$$d = -\min_{i=1,..,n}[\ \mathrm{sdist}(\ \mathrm{comp}(MS_i(\mathbf{p},0))\ ,\ S_i(\mathbf{p}_i)\ )\ ]. \qquad \mathrm{C1}$$

This contains variables $\mathbf{p}$, $\mathbf{p}_1$, $\mathbf{p}_2$, ..., $\mathbf{p}_n$. The signed distance expressions all involve simple features. It can be assumed that these expressions can be expressed in terms of DOF-variables.

The SUPINF-algorithm can be used to perform stage 2 of the calculation. We need to find the minimum of the above expression for d in terms of $\mathbf{p}_1$, ..., $\mathbf{p}_n$ while letting $\mathbf{p}$ vary. This is given by,

$$\mathrm{INF}(\ d,\ \{\},\ \{\mathbf{p}_1,...,\mathbf{p}_n\}\ ), \qquad \mathrm{C2}$$

where d is given by C1. Unfortunately SUP and INF do not work well if their first argument is a complicated expression and their second argument contains few or no constraints. It is possible to turn things round by introducing a variable, τ, satisfying the following constraints to be denoted $C\tau$:

$$\left.\begin{array}{l} \tau \geq -\mathrm{sdist}(\ \mathrm{comp}(MS_1(\mathbf{p},0))\ ,\ S_1(\mathbf{p}_1)\ ), \\ \qquad ..., \\ \tau \geq -\mathrm{sdist}(\ \mathrm{comp}(MS_n(\mathbf{p},0))\ ,\ S_n(\mathbf{p}_n)\ ), \end{array}\right\} C\tau$$

Then C2 can be rewritten as

$$INF(\tau, C\tau, \{\mathbf{p}_1,\ldots,\mathbf{p}_n\}). \qquad C3$$

The result is an expression involving $\mathbf{p}_i$ (i=1,...,n), but not $\mathbf{p}$. The expression represents the minimum size of the measuring solid in terms of $\mathbf{p}_i$ - the positions of the zones.

In stage three of the calculation it is assumed that the $\mathbf{p}_i$ satisfy constraints Cz. The result of this calculation is the maximum required size of the measuring solid and will be denoted d_{max}. It is the SUP of C3 over Cz:

$$d_{max} = SUP(INF(\tau, C\tau, \{\mathbf{p}_1,\ldots,\mathbf{p}_n\}) , Cz , \{\}). \qquad C4$$

The SUP ensures that we find the maximum size that can occur due to variation in position of the zones whereas the INF ensures that the position of the measuring solid is chosen to minimise its size.

STAGE 2: The Constraints on the Measuring Solid Relative to the Least Material Solid

Having found the maximum size for the measuring solid we can move on to stage two of the derivation of constraints between a datum of a composite feature and a zone of one of the simple features. Use is made of the fact that the material region of each zone must be enclosed by the measuring solid.

The relationships between the zones and the datum can be considered individually and so we need only be concerned with a single zone in the analysis presented here. Let $Mat_i(\mathbf{q})$ denote the space occupied by the material region (the least material solid) of any of the zones of simple feature i when its position is $\mathbf{q}$.

The signed distance between the complement of the measuring solid and $Mat_i(\mathbf{q})$ must be greater than or equal to zero. To obtain the most general constraints the measuring solid should have its maximum size, d_{max}, giving the constraint,

$$sdist(\ Mat_i(\mathbf{q})\ ,\ comp(MS_i(\mathbf{p},d_{max}))\ ) \geq 0.$$

This expression involves variables representing the positions of the the zone and datum. However, the constraints applied to the relationship between the zone and datum must involve the DOF-variables of that relationship. The vector of these DOF-variables, to be denoted $\mathbf{x}$, is given by $\mathbf{x}=\mathbf{p}-\mathbf{q}$.

Suppose that the zone and the measuring solid are both given a translation of $-\mathbf{q}$. Then, the above expression will continue to hold. Therefore, it follows that

$$sdist(\ Mat_i(\mathbf{0})\ ,\ comp(MS_i(\mathbf{x},d_{max}))\ ) \geq 0.$$

This is the constraint to be applied to the relationship between the datum and the zone of feature i.

The constraints thus obtained put an upper bound on the set of possible relative positions of each zone and the datum but there is no guarantee that they describe the set exactly. Although for some features it can be shown that the constraints do represent the worst possible case, in general this is not true. However, they are sufficient in the absence of any stricter constraints.

-----oOo-----

In conclusion, there are three ways of representing composite features in the zone-datum network.

- The first is applied if each simple feature is toleranced individually.

- The second is applied if a tolerance has been given to the whole composite feature and if it is satisfactory for the zones of the composite feature to appear as single nodes in the network.
- The third is also applied if a tolerance has been given to the whole composite feature but if it is desirable for the individual simple features to appear explicitly in the network.

It was explained how constraints can be attached to each of the arcs in the networks of each of these representations. In many cases the techniques described for simple features are applicable. There are two major complications. Firstly, zones are split into subzones in representation 3. In this case there is an additional type of relationship occurring between the subzones that make up a composite zone.

Secondly, constraints must be associated with the arcs in representation 1 which lead to the datum defined by the feature. The problem is that the features are toleranced independently but the position of the datum is affected by the surface of each feature.

5.11. OBTAINING RESULTS FROM THE REPRESENTATION

This section shows how results can be obtained from analysis of the zone-datum network. The total constraints on a pair of nodes are found by combining constraints in the network.

The zone-datum network derived in this chapter is analogous in many ways to the network of nominal parts derived in chapter 3. In both cases the arcs represent constraints on the relative positions of the nodes. The constraints take the form of inequalities.

The networks of this chapter and of chapter 3 represent different physical situations, however. In chapter 3 the constraints represented bounds on motion between the parts. If a force were applied to one part in the structure then the parts would accommodate to that force: more than one part might move and the positions attained by the parts would be such that they give maximum displacement to the point at which the force was applied.

On the other hand, the physical situation in this chapter consists of features whose positions are fixed in a single part: they are determined at the time of manufacture and are subject to whatever random variations occurred at that time. Suppose that each manufacturing process in the production of the part is completely independent of all others. The position of a certain feature might be dependent on several manufacturing processes. Such a feature would attain maximum deviation only if all the processes produced results with maximum deviation from their nominal value. Although this is possible it has a low probability. Therefore, the probability distribution of the location of a feature may not be uniform. This contrasts with the situation in chapter 3 where it is likely that a part will attain its maximum deviation.

Despite this it is useful to determine worst possible cases for the locations of features in a part. Therefore, analysis of the sort used in chapter 3 is used again here to find bounds on ranges.

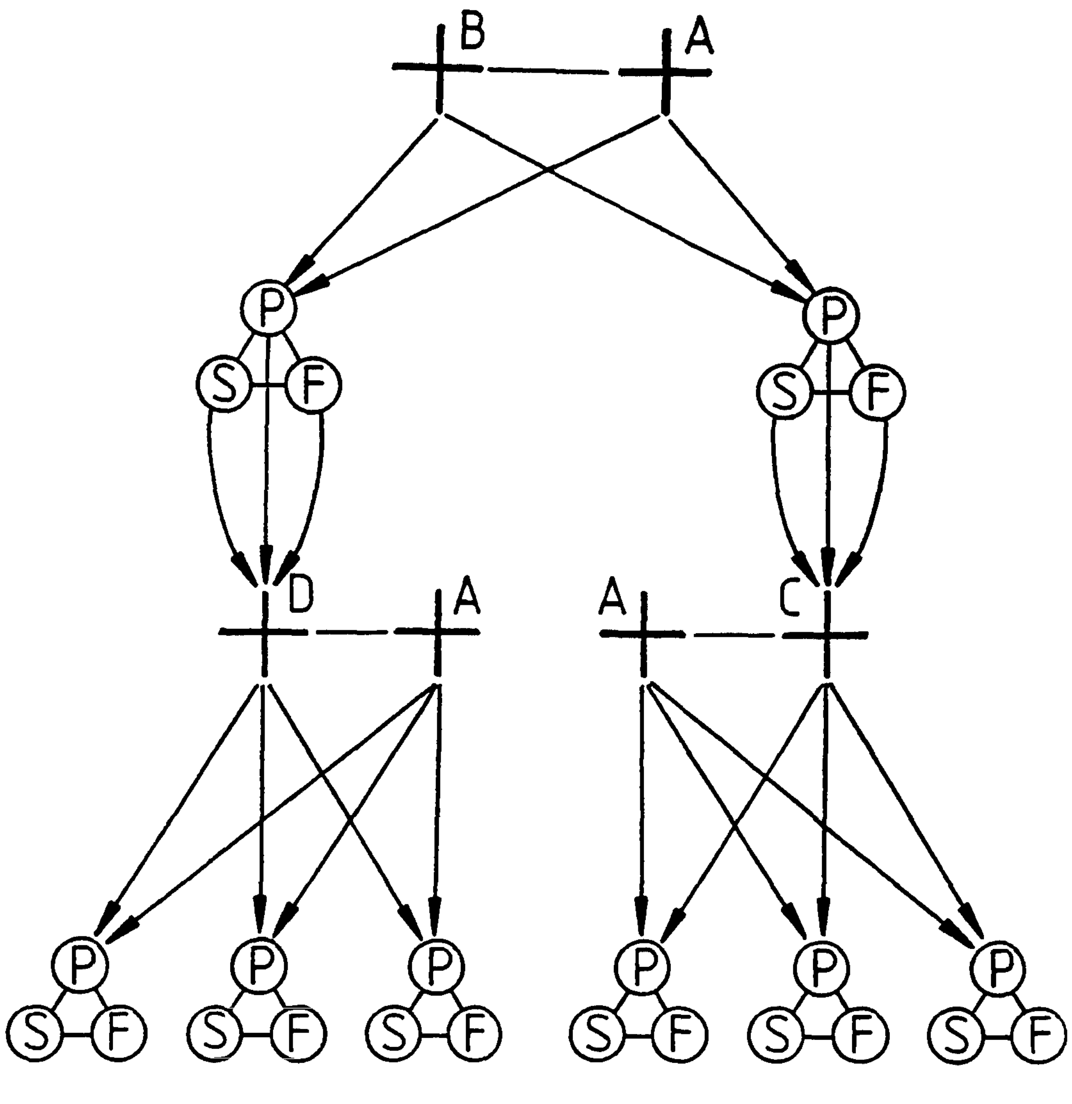
There are two types of inference illustrated diagrammatically in figure 3.1.1. The first takes a set of parallel arcs between two nodes and replaces them by a single arc containing constraints expressing the total effect of the constraints in the original arcs. The second inference takes three nodes, A, B and C with an arc between A and B and an arc between B and C. The inference finds constraints on C relative to A and attaches these to a new arc between A and C. Using these two inferences it is possible to find the constraints on the relative position of any two nodes which take into account all relationships between zones and datums.

The order of application of the inferences is determined as described in section 3.1 under "Application of the Inferences". Basically all paths between two chosen nodes are found. The constraints implied by each path are combined using repeated application of the second inference described above. The results from each path are combined using the first inference.

Reducing the Number of Paths to be Analysed by Considering the order of Manufacture of Features

It is useful to have means by which the number of paths to be analysed can be reduced. One way to do this assumes that the order that features were produced during manufacture can be determined from the zone-datum network. For example if hole B is located relative to a datum defined by hole A then it follows that A must have been produced before B. A path cannot imply constraints if it involves relationships which only existed after the path's terminal features had been produced. Put another way, the positions of two features cannot be affected by features which do not yet exist.

A causal direction can be associated with certain arcs in the network to show the order in which the features have been produced. This can be done for all arcs which link a zone to a datum. The arc should be directed to show which came first during manufacture. When the datum is being used to locate the zone the direction should be from the datum to the zone. When the zone defines the datum the direction should be from the zone to the datum. Arcs between two datums or between two zones are left undirected. Figure 5.11.1 shows the network of figure 5.9.16 with directions attached to appropriate arcs.



The zone-datum network shown previously in figure 5.9.16 for the part in figure 5.1.1 with arcs given a direction according to their order of manufacture.

Figure 5.11.1

For example, suppose we want to find constraints on the positions of the large holes in the part shown in figure 5.1.1 with this network. These holes were cut after the two planes defining datums A and B but before any of the six small holes. Therefore the positions of the large holes could not have been affected by the positions of the small holes though they could be affected by the positions of the planes defining A and B. As a result it is not necessary to consider any of the paths between the large holes which go via the nodes representing the small holes. We shall say that there is no causal explanation for constraints along such paths.

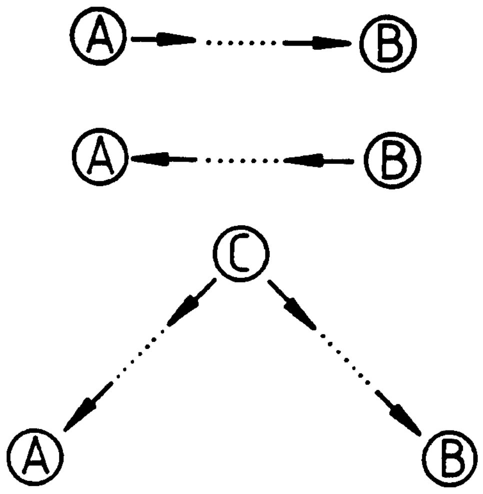
Note that all zone-datum networks in this thesis have been drawn such that directed arcs are in the downwards direction. This allows us to say that one node is "above" (or "below") another if all directed arcs between them point in the direction from the first to the second (or second to first).

A path between two nodes A and B must not go lower than both of A and B, since otherwise it would be passing through nodes associated with features that did not exist when the positions of A and B were determined.

The same rule must apply to any section of the path. If nodes C and D lie on the path then the section of the path between C and D must not go lower than both of C and D. Therefore, the positions of two nodes A and B are constrained only by paths with one of the following properties which are illustrated in figure 5.11.2.

- All directed arcs in the path point in the direction A to B;
- All directed arcs in the path point in the direction B to A;
- The path contains a node C such that all directed arcs between A and C have direction C to A and all directed arcs between C and B have direction C to B. (In this case the path constrains the relative position of A and B because both are constrained relative to C.)

If the order of manufacture of features is known then only paths with one of these properties need be considered when finding the constraints on nodes A and B.



Paths which put constraints on A and B must contain arcs with one of these causal orderings.

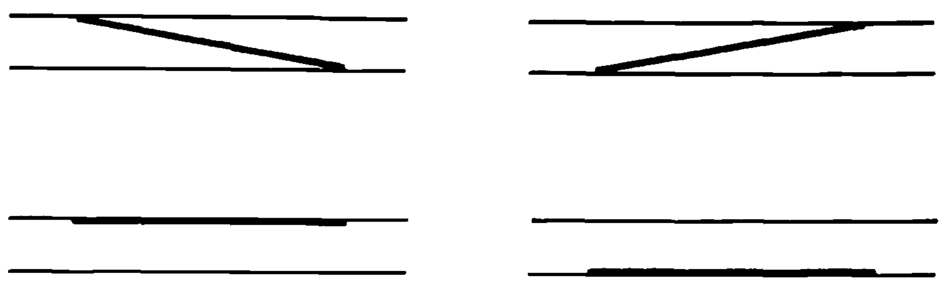
Figure 5.11.2

Obtaining Useful Results

So far it has been shown how constraints can be found on the positions of zones and datums. Since these do not exist in a real part these results are not directly useful. It is more useful to find the positions of real features contained in the zones. This depends on how the position of a real imperfect feature is measured. Some suggested methods are the following.

- Firstly, it might be acceptable to assume that the form of the surface is perfect when measuring its position. If this is so then the result of such a measurement could be predicted by adding the variation in position of the surface inside the zone to the variation in position of the zone.

For example, figure 5.11.3 shows the greatest inclination and translational displacements of a perfect real feature inside a planar zone. The assumption that the feature has perfect form avoids the question of how the position of an imperfect feature can be defined. It will be an acceptable approximation in many circumstances.



Extremes of inclination and translational displacements for a perfectly formed surface inside a zone.

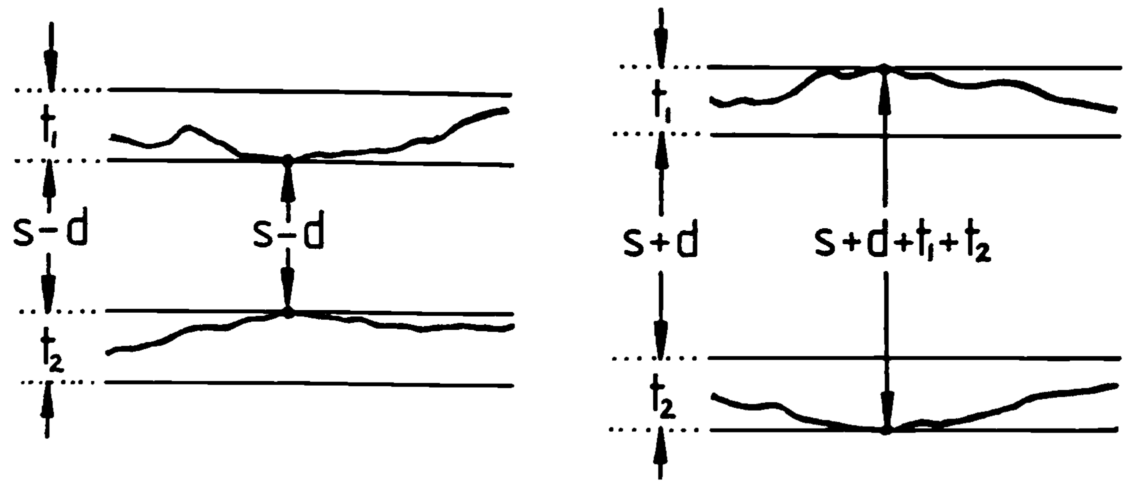
Figure 5.11.3

- Secondly, the position of a feature could be measured by resting some surface against it and measuring the position of this surface.

The result of this could be predicted by letting the feature define a datum. The variation in position of the datum is taken as the variation in position of the feature. This assumes that the position of a datum defined by a feature is the same as the position of a surface resting against the feature.

- Thirdly, the distance between two surfaces could be measured at single points for example by using calipers whose arms touch the surfaces at single points.

The result of such a measurement could be predicted by adding the thicknesses of the zones to the variation in position of the zones. For example, suppose two parallel planar zones are separated by a distance of $s \pm d$ and that they have thicknesses t_1 and t_2 (figure 5.11.4). Then the distance between surfaces contained in the zones measured at single points varies between $s-d$ and $s+d+t_1+t_2$.



The variation in distance between two surface points contained in two zones whose separation is variable.

Figure 5.11.4

The real feature may be contained in more than one tolerance zone. Each zone implies different constraints on the position of the real feature. The total constraints on the position of the feature is a combination of these constraints.

5.12. CONCLUSION

This chapter has shown how a geometrically toleranced part can be represented in a computer. The geometry of relationships between tolerance zones and datums has been analysed. Parts with unknown and variable shape are represented by datums and tolerance zones both of which are ideal geometric objects. A zone-datum network is produced with arcs representing constraints between zones and datums. The representation allows each feature of the part to have tolerances of form, size, orientation, absolute position and MMC-position as defined by Requicha.

An understanding has been developed of the geometry of relationships between tolerance zones and datums. Two of the most difficult problems were deciding (1) how a datum is located relative to a tolerance zone of its associated feature and (2) how zones of the same feature are located relative to one another. The possibility of some features being composite made these problems more difficult.

The derivation of constraints between zones and datums is made by two types of technique both of which would be straightforward to implement. Firstly, the possible sets of constraints for a given type of relationship could be stored in a catalogue. In the relationships between datums in a datum-system and between a datum and a zone located with respect to it there are only a small number of possible situations and so they can be catalogued easily.

Secondly, some constraints are derived using signed distances. A catalogue could be available of signed distance expressions for different shapes of feature. In most cases signed distance expressions only have to be known for simple features. Constraints on relationships involving composite features can usually be evaluated using signed distance expressions for their component simple features.

Derivation of constraints using signed distances as described in this chapter does not always produce constraints that are realistic. However, the constraints are useful because they provide a bound on the set of positions of the items which are being constrained. It is not obvious how to obtain techniques that provide stronger, more realistic constraints that can be applied to any type of feature.

The network of zones and datums can be analysed to answer questions about the extreme relative position of features. The representation of a part is similar to the representation of an assembly of perfectly formed parts described in chapter 3.

Because of this, the methods for obtaining results are also similar.

Chapter 6: **ASSEMBLIES OF TOLERANCED PARTS**

Introduction

Given an assembly of toleranced parts there are many interesting questions that are useful to answer. For example it is important to know whether the parts of an assembly will fit together no matter what instances of the parts are chosen for assembly. Other questions involve the maximum or minimum slop or maximum displacement attainable by a part. In general there are many ways that the shapes of parts can vary and it is not easy to find the worst case over all variations.

Use is made of concepts from previous chapters: assemblies of parts from chapter 3 and toleranced parts from chapters 4 and 5. As a result, there are two types of variation. Firstly there is variation in positions of parts (dealt with in chapter 3). The variables which represent this variation are called sloppy variables. They are the DOF-variables introduced in chapter 3 and are denoted by a capital Δ prefix.

Secondly, there is variation in the shape of parts (dealt with in chapters 4 and 5). This is represented by rigid variables which describe the positions and sizes of the zones and datums that make up the part. They are called rigid variables because they describe possible variations between instances of a rigid structure. Rigid DOF-variables are denoted by a small δ prefix.

Variation represented by sloppy variables occurs within each instance of the assembly produced whereas variation represented by rigid variables occurs between different assembly instances. (An assembly instance is created by choosing an instance of each of the parts and assembling them.) The shape of each part is constant in a single instance but is variable if many instances are considered. Each part in an assembly instance has a set of possible positions and this set is different in different instances of the assembly. It follows that a set of sets of positions can be associated with each part.

Constraints are represented by a set of inequalities involving both types of variable. Usually, it is possible to divide the inequalities into two categories,

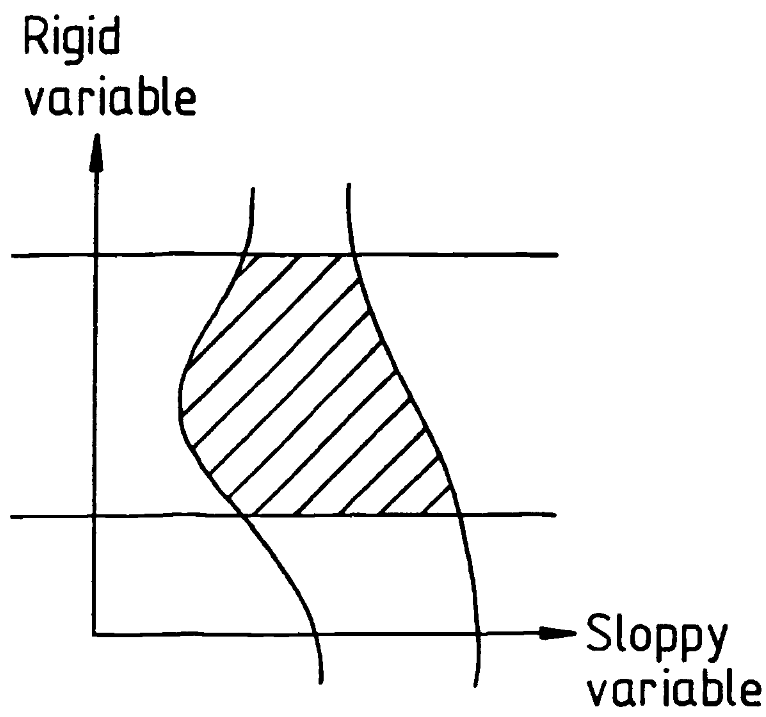
- inequalities involving sloppy variables and rigid variables and
- inequalities involving only rigid variables.

Each solution to the latter inequalities represents a possible assembly instance. For each of these solutions the former inequalities have a different solution set over the sloppy variables. A solution of the former inequalities represents a set of configurations of the assembly. Thus the complete set of inequalities represents a set of sets of configurations of the assembly.

A simple graph may be drawn, as shown in figure 6.0.1, with axes to represent sloppy and rigid variables. Since the graph is two-dimensional only one sloppy variable and one rigid variable can be included. Inequalities containing only the rigid variable have a solution set represented by a horizontal band on the graph. The solution to inequalities which include both types of variable is a region bounded by curved lines.

Certain assumptions, listed below, have been made about parts and assemblies of parts. These assumptions were specified in the introduction to chapter 3.

- All parts are rigid.
- Parts are not glued or bolted to one another in any way.
- The nominal position of each part is known.
- The parts deviate only by short distances from their



The representation of the solution to inequalities over rigid and sloppy variables as a graph. The horizontal lines bound a region representing the solution to inequalities involving only rigid variables. The curved lines bound a region representing the solution to inequalities with both rigid and sloppy variables.

Figure 6.0.1

nominal positions.

- The parts are not subject to gravity or any other force and therefore their positions are constrained only by their contacts with other parts.
- There is a set of potential contacts between pairs of features which allow the features to be in contact or to separated by a small amount. Each potential contact acts between regions of two features and due to the small uncertainty assumption the size and shape of such regions are known to a good approximation.

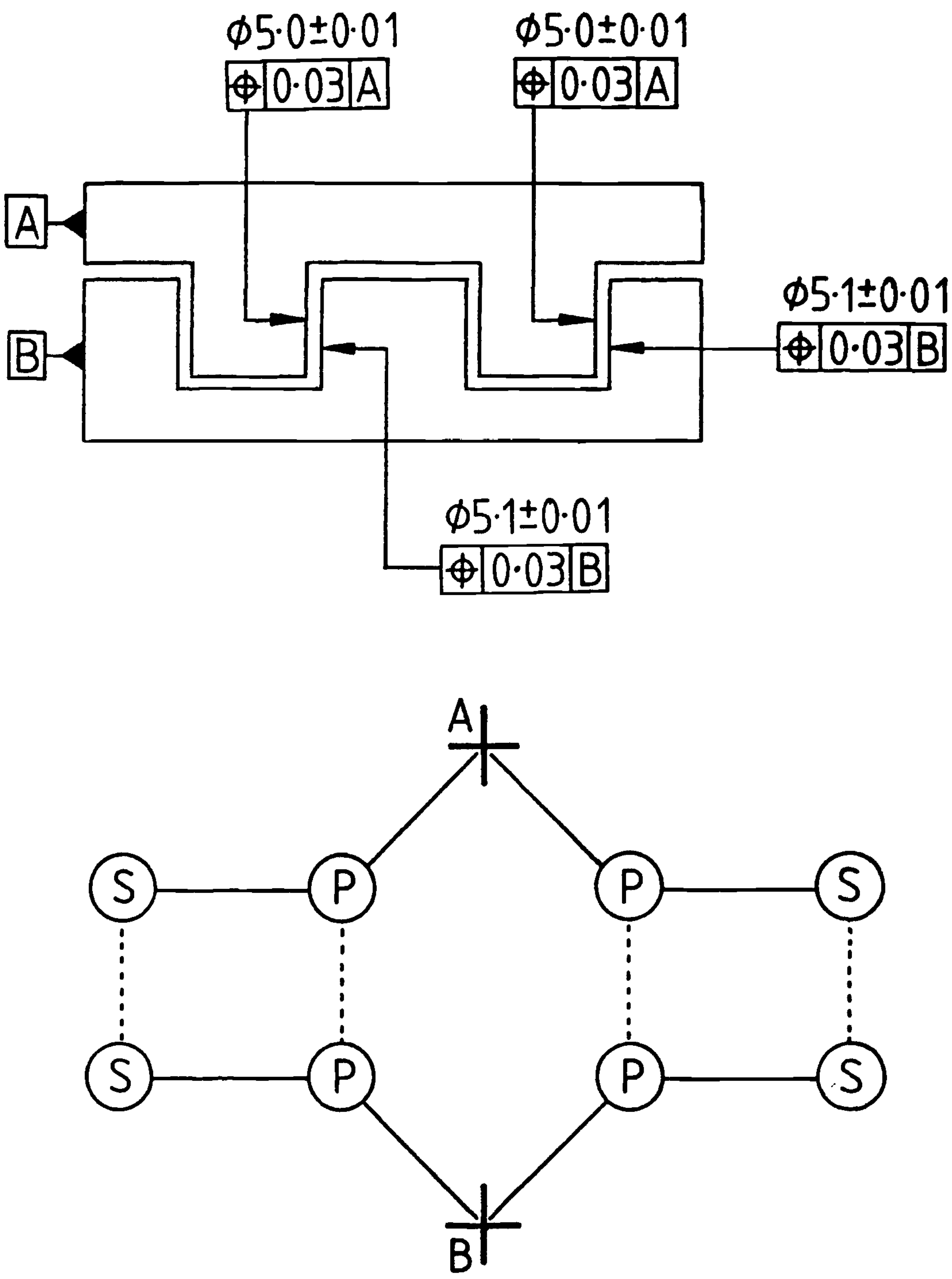
An additional assumption made about assemblies of toleranced parts is that during the process of assembly the parts are selected randomly. This means that the shape of a part is completely independent of all other parts. Therefore, two parts at opposite extremes of their tolerance range might end up in the same assembly. For example a peg with maximum possible diameter might be inserted into a hole with minimum possible diameter or vice versa. The tolerance specification must be strict enough for these extremes to be functional. (An alternative method of assembly would be to choose parts with the goal of attaining as good a fit as possible. Although this would allow better fits between parts to be obtained it is not much used in practice. Hence, this type of assembly has not been dealt with in this thesis.)

Introduction to the Zone-Datum Network of an Assembly

An assembly of parts is to be represented as a large network of zones and datums. Each part creates its own network of zones and datums. These networks are linked by relationships between zones whose associated real features have a potential contact between them. Section 1 of this chapter formalises these relationships.

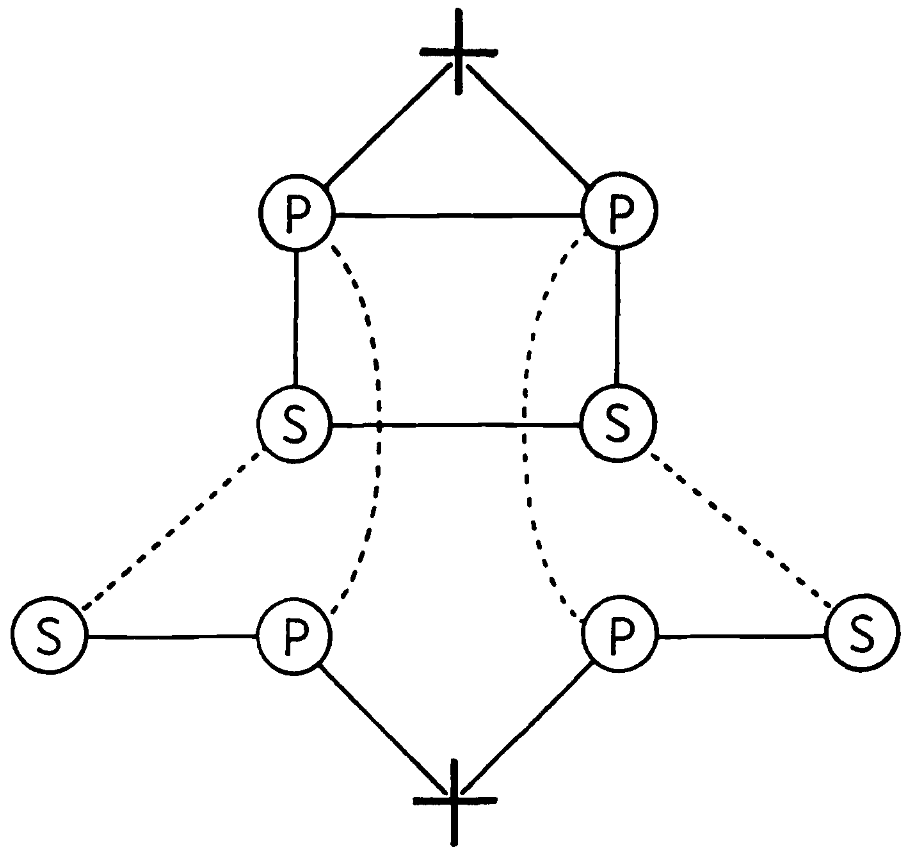
Representing assemblies of parts in this way means that they can be analysed in the same way as single parts. Analysis of the network proceeds in a similar way to that described in the last chapter. The constraints that result can be used to answer questions about the assembly.

A simple assembly is shown in figure 6.0.2 along with its zone-datum network. Both pegs of the upper part have a position tolerance with respect to datum A and both holes of the lower part have a position tolerance with respect to datum B. On top of this, each peg and hole has a size tolerance. Each peg is potentially



A simple assembly and its zone-datum network. Dashed lines indicate relationships associated with potential contacts.

Figure 6.0.2



The zone datum network of the above assembly if the two pegs of the upper part were toleranced as a single composite feature.

Figure 6.0.3

in contact with one of the holes. For simplicity, contact between the horizontal plane surfaces is ignored.

The zone-datum network of the assembly is shown with solid lines to represent relationships between zones and datums of the same part and dashed lines to represent relationships between zones of different parts (the ones of interest here). Note that, for each potential contact, there are two of these relationships, one between the position tolerance zones and one between the size tolerance zones.

Composite features can be incorporated into this representation. Suppose that the two pegs on the upper part are defined as a composite feature to which a position tolerance and a size tolerance are applied but that the holes of the lower part are separate simple features. The feature allocation of the two parts are now incompatible: the composite feature makes a potential contact with each of the simple features. Using representation 3 of composite features (figure 5.10.2(iii), section 5.10) the situation can be represented by the network in figure 6.0.3. The nodes, of the upper part correspond to subzones of the individual pegs and are linked by relationships which indicate that they have fixed positions relative to one another.

6.1. CONSTRAINTS FROM CONTACTS

When two zones contain real features that are potentially in contact their positions are constrained. This section formalises such constraints and shows how they can be represented. Account has to be taken of the fact that there is an infinite number of different surfaces that can be contained in the zones and that each possible pair of surfaces constrains the zones in a different way.

An Example

To start with a simplified example will be given. After that a more formal analysis of the derivation of the constraints between zones containing features potentially in contact will be given. The example will aid visualisation of the problem.

Consider two planar features potentially in contact. Their tolerance zones (figure 6.1.1(i)) are defined as

$$O(m_1;H_1') - O(l_1;H_1') \quad \text{and} \quad O(m_2;H_2') - O(l_2;H_2')$$

where H_1' and H_2' are copies of the extended features of the two features. Offsets m_1 and m_2 are usually positive and l_1 and l_2 are usually negative. Coordinate systems are located in the zones in the standard way (section 3.1 under "Features") except that for convenience the two coordinate systems are given the same orientations. The goal is to find constraints on the relative positions of the zones.

For simplicity, in this example, only one degree of freedom Δx is considered and is measured in the direction normal to the surfaces of the zones. Δx is zero when the origins of the coordinate systems are coincident.

Another simplification is that the real feature is assumed to be a perfect plane parallel with the nominal feature but with variable position in the x-direction.

For any such surfaces, (see figure 6.1.1(i)), there is some number c such that

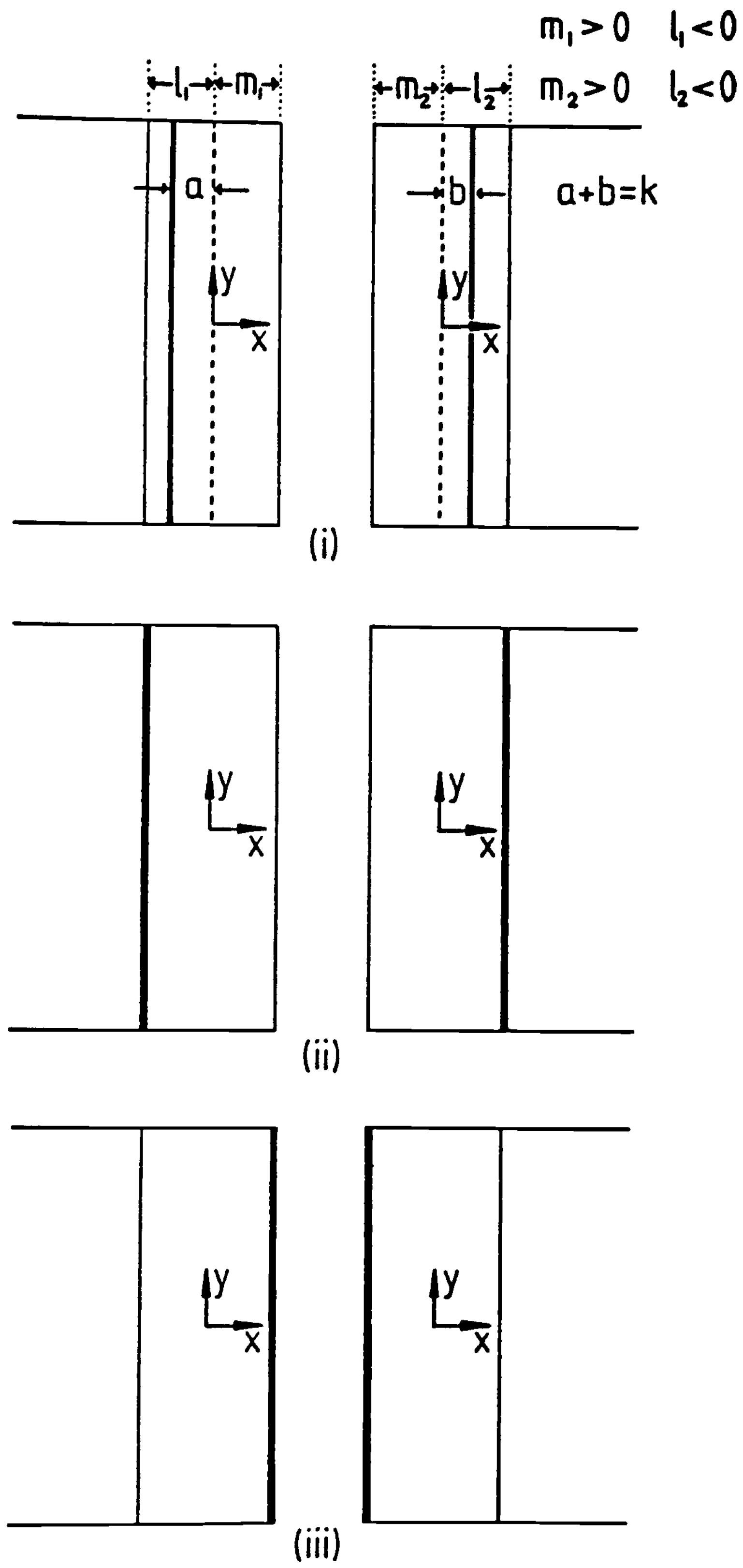
$$\Delta x = c$$

when the features are in contact. Hence a potential contact constrains Δx by an inequality of the form

$$\Delta x \geq c.$$

Figures 6.1.1(ii) and (iii) show cases where the real features are coincident with the boundaries of the zones. From these situations, extreme values of c can be predicted by considering the value of Δx when the surfaces are in contact. Simple analysis of the geometry in figures (ii) and (iii) shows that the following hold,

$$m_1+m_2 \geq c \quad \text{and} \quad c \geq l_1+l_2.$$



Two zones of planar features containing real features which are potentially in contact. (i) Real features are at arbitrary positions. (ii) and (iii) Real features are at extreme positions.

Figure 6.1.1

Hence, c is bounded above by m_1+m_2 and below by l_1+l_2. The complete set of constraints between the two zones can be written,

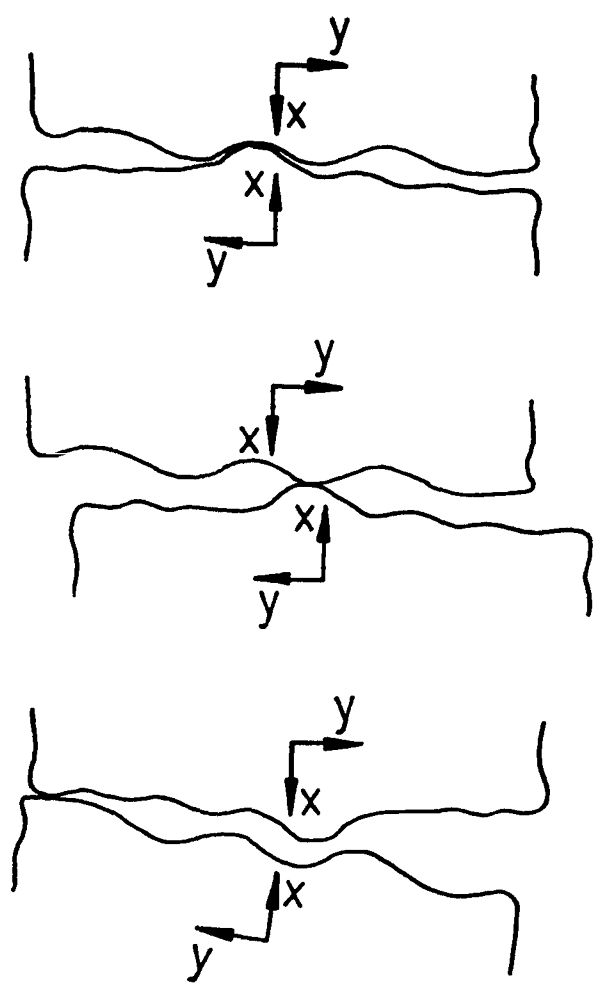
$$\Delta x \geq c, \quad m_1+m_2 \geq c \geq l_1+l_2. \qquad \text{C1}$$

Note that c is like a rigid variable since it has a fixed value in each instance of the assembly whereas Δx is variable in each instance. For each value of c there is a different range of Δx. However, it will be shown below that, in general, c must be replaced by a function of sloppy variables and then must be treated as a different type of entity.

The General Case

The above example was restricted in that only one degree of freedom was involved. In general, all degrees of freedom are constrained and interact so that the inequalities describing the constraints between the zones must involve all degrees of freedom. The constraints between two zones containing surfaces in contact have a form similar to constraints C1 but with two generalisations. Firstly, Δx is replaced by an expression involving many DOF-variables.

Secondly, c is replaced by a function with sloppy DOF-variables as arguments. This function reflects the effect of the uneven surfaces of the features. Figure 6.1.2 shows that when rotations and all translations are allowed then the lower bound of Δx is dependent on these degrees of freedom. This is because the point of contact is not uniquely determined. The function expresses Δx in terms of other sloppy variables. Although the function can never be known precisely due to the potentially complex shape of the surfaces it satisfies properties that make it easy to handle.



Two real features in contact. Translation in the y-direction and rotation both affect the x-component of the positions of the features when they are in contact due to the uneven surfaces.

Figure 6.1.2

Unfortunately it is not convenient in general to express the function which replaces c in terms of standard degrees of freedom. Therefore a set of more convenient non-standard degrees of freedom is introduced.

The main requirement of the new degrees of freedom is that there is a variable which always increases as contact between the surfaces is broken. The new degrees of freedom are denoted $\Delta\mu$, $\Delta\lambda_1$ and $\Delta\lambda_2$. (There are only three new degrees of freedom because the standard rotational degrees of freedom, $\Delta\theta$, $\Delta\phi$ and $\Delta\psi$ are to be used as well.) The new degrees of freedom are translations but need not be orthogonal. They can be expressed in terms of standard DOF-variables ($\Delta x,\ldots\Delta\psi$) but the relationship between the two sets of variables depends on the type of features (planar, cylindrical etc.) that we are dealing with.

The new DOF-variables are chosen as follows. First, $\Delta\mu$ must have the property that if the **nominal** features are in contact then increasing $\Delta\mu$ moves the parts in the direction of the normal of the surfaces taken at the point of contact. Thus increasing $\Delta\mu$ breaks the contact. Next, $\Delta\lambda_1$ and $\Delta\lambda_2$ are chosen so that $\Delta\mu$, $\Delta\lambda_1$ and $\Delta\lambda_2$ can be used to uniquely specify any position ignoring rotation. $\Delta\mu$, $\Delta\lambda_1$ and $\Delta\lambda_2$ should be expressed using standard translational DOF-variables and not rotational DOF-variables.

Three examples of contacting features and their associated non-standard DOF-variables are given here. Firstly, in the case of planar features with their x-axes pointing out of the planes $\Delta\mu$ is simply equivalent to Δx. This is because the planes can be separated by increasing Δx. Then, so that $\Delta\mu$, $\Delta\lambda_1$ and $\Delta\lambda_2$ are a complete set of translational degrees of freedom, $\Delta\lambda_1$ and $\Delta\lambda_2$ are chosen to be equivalent to Δy and Δz.

Secondly, consider the case of a cylindrical peg in a cylindrical hole with x-axes pointing along their axes. $\Delta\mu$ is $-\sqrt{\Delta y^2+\Delta z^2}$. The minus sign in front of the square root is necessary so that $\Delta\mu$ increases as the surfaces are separated. $\Delta\lambda_1$ and $\Delta\lambda_2$ can be chosen as, $\mathrm{sign}(\Delta y)\tan^{-1}(\Delta y/\Delta z)$ and Δx.

Thirdly, contact between a tab in a slot with x-axes normal to their faces is broken by motion in one of two directions depending on which faces of the tab and slot are in contact. $\Delta\mu$ chosen as $-|\Delta x|$ has the correct property. However, since for each value of $\Delta\mu$ there are two values of Δx there is no easy way to choose $\Delta\lambda_1$ and $\Delta\lambda_2$ so that any position can be defined uniquely. It is necessary to introduce a new parameter, $\Delta\lambda_3$, which has value

1 when Δx is positive or zero and value -1 when Δx is negative. $\Delta\lambda_1$ and $\Delta\lambda_2$ can be chosen as Δy and Δz.

In the discussions in this chapter the possible existence of $\Delta\lambda_3$ will usually be ignored because it is straightforward to introduce $\Delta\lambda_3$ where necessary.

Now that these degrees of freedom have been defined they will be used to explain the constraints on the positions of two zones containing features in contact. Define $\boldsymbol{\lambda}$ to be the vector of five DOF-variables,

$$\boldsymbol{\lambda} = (\Delta\lambda_1, \Delta\lambda_2, \Delta\theta, \Delta\phi, \Delta\psi).$$

Then, for any pair of features in contact it is possible to find some function, c, such that

$$\Delta\mu = c(\boldsymbol{\lambda}). \qquad \text{C2}$$

This function corresponds to the variable c in the example given earlier. In that example c was the value of Δx at contact. In general, however, there is no unique value of Δx at contact because it is dependent on the value of other degrees of freedom. In the example there was a different value of variable c for each possible pair of surfaces and, correspondingly, there is a different function c for each possible pair of surfaces. In general, such functions will be called c-functions. Since the exact shape of real features is unknown the exact behaviour of a c-function is also unknown.

Since $\Delta\mu$ increases as the features are separated it follows that the **potential** contact is described by the constraint,

$$\Delta\mu \geq c(\boldsymbol{\lambda}). \qquad \text{C3}$$

In the case of a tab in a slot $\Delta\mu = -|\Delta x|$ and there are two pairs of contacting surfaces. Since there are two values of $\Delta\lambda_3$, C3, in this case, actually represents two inequalities,

$$c(\Delta\lambda_1, \Delta\lambda_2, -1, \Delta\theta, \Delta\phi, \Delta\psi) \leq \Delta x \leq -c(\Delta\lambda_1, \Delta\lambda_2, 1, \Delta\theta, \Delta\phi, \Delta\psi).$$

It will now be shown that there are constraints on c-functions with a form similar to the constraints on the variable c in the earlier example.

For a given feature there is a class of possible surfaces satisfying its tolerance specification. Each pair of possible surfaces in contact gives rise to a different function, c. Hence there is a class of possible c-functions. The derivation of constraints on c-functions requires consideration of two typical tolerance zones containing features in contact. First, some notation is introduced.

Let F_1 and F_2 be nominal features with extended features H_1 and H_2. Let Z_1 and Z_2 be two zones defined by

$$Z_1 = O(m_1;H_1') - O(l_1;H_1') \quad \text{and} \quad Z_2 = O(m_2;H_2') - O(l_2;H_2'),$$

where H_1' and H_2' are copies of H_1 and H_2. The features have extent-solids E_1 and E_2.

The <u>solid of minimum material</u> of a zone is defined as the material region of the zone intersected with the extent solid of the feature. It represents a region from which material cannot be removed (in the vicinity of the significant portion of the zone) without generating a surface which violates the tolerance specification.

Conversely, the <u>solid of maximum material</u> of a zone is equal to the complement of the air region of the zone intersected with the feature's extent solid. It represents a solid to which material cannot be added without violating the tolerance specification (in the vicinity of the significant portion of the zone at least). M_1, M_2, L_1 and L_2 are illustrated for a planar feature in figure 6.1.3 and can be expressed as,

$$M_i = O(m_i;H_i') \cap E_i \quad \text{and} \quad L_i = O(l_i;H_i') \cap E_i, \quad (i=1,2).$$

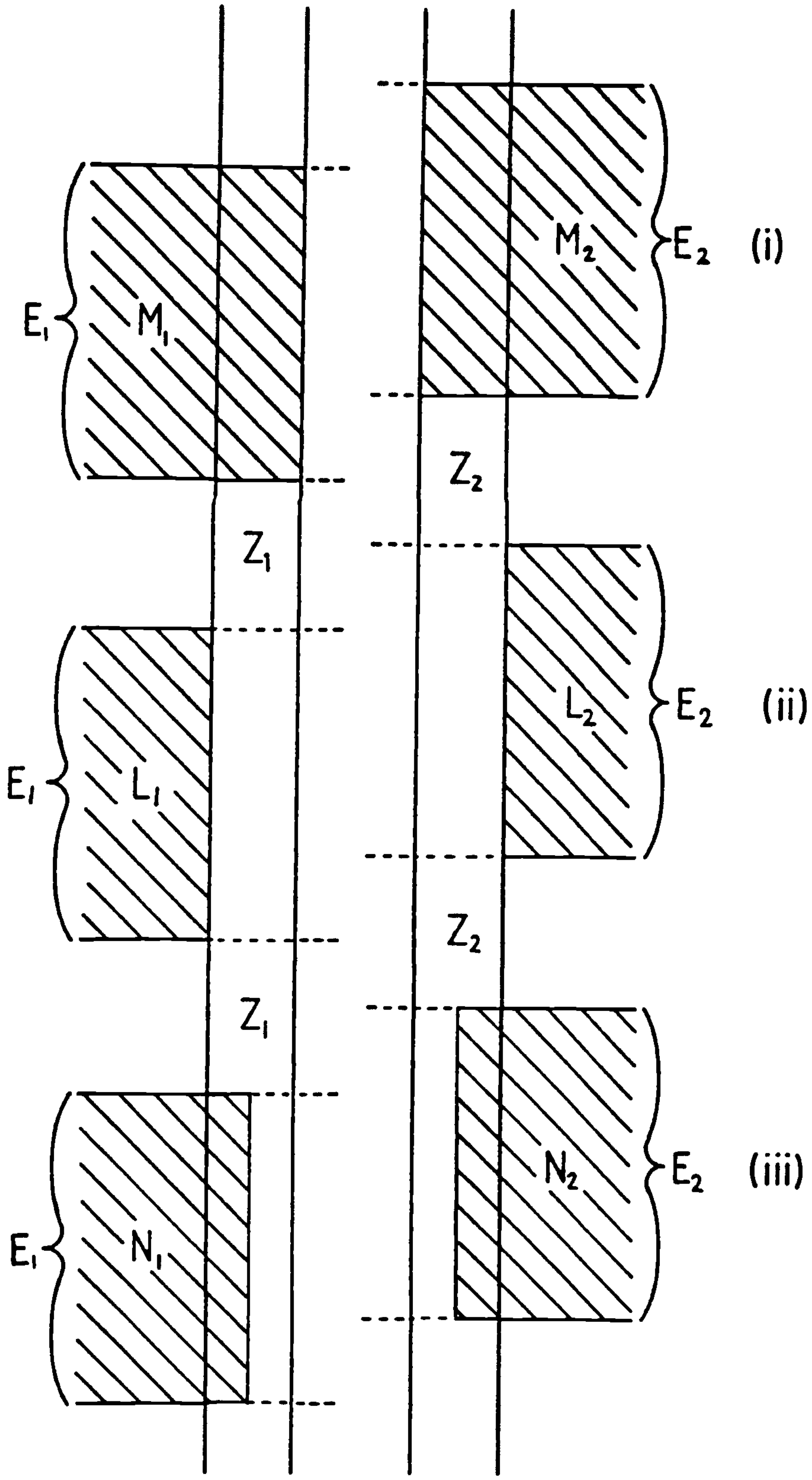
Also define solids, N_1 and N_2, shown in figure 6.1.3, part of whose surfaces are coincident with the extended features and which are intermediate between the maximum and minimum material solids. These parts of their surfaces are congruent with the nominal feature.

$$N_i = H_i' \cap E_i, \quad (i=1,2).$$

The constraints on the positions of the zones can be written in terms of signed distances (cf. the relationships involving inequalities in section 5.9.3 and 5.9.4). It can be assumed that an expression for signed distance between N_1 and N_2 is known (see appendix 1). The signed distance expression depends on the nominal overlap of the features. Suppose that the relative position of two zones is given by $\Delta\mu$ and λ. Then, define <u>sdist</u>$(\Delta\mu,\lambda,N_1,N_2)$ to be the signed distance between N_1 and N_2 when the position of the coordinate systems in the zones is given by $\Delta\mu$ and λ.

It is also possible to define signed distances between L_1 and L_2 and between M_1 and M_2, sdist$(\Delta\mu,\lambda,L_1,L_2)$ and sdist$(\Delta\mu,\lambda,M_1,M_2)$. Again $\Delta\mu$ and λ are measured with respect to the coordinate systems in the zones. When rotations are small the three signed distance expressions are related by the following, as shown by examination of figure 6.1.4,

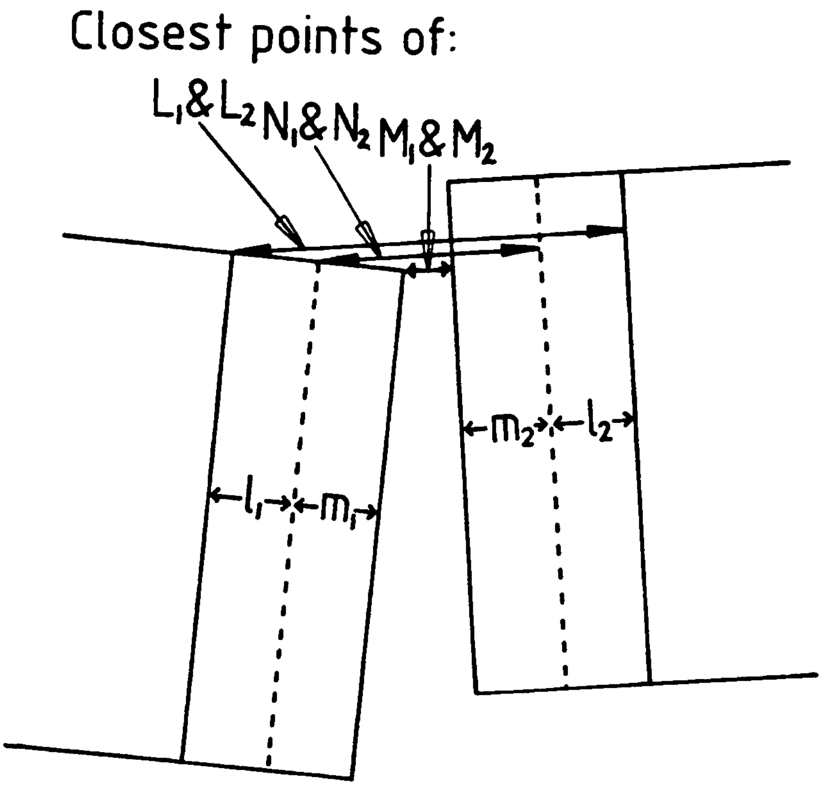
$$\text{sdist}(\Delta\mu,\lambda,L_1,L_2)+l_1+l_2$$
$$= \text{sdist}(\Delta\mu,\lambda,N_1,N_2) =$$
$$\text{sdist}(\Delta\mu,\lambda,M_1,M_2)+m_1+m_2,$$



(i) Maximum material solids.
(ii) Minimum material solids.
(iii) Intermediate solids part of whose surfaces are coincident with H_1' and H_2'.

Figure 6.1.3

where m_1, l_1, m_2 and l_2 are the offsets used in the definition of the zones.



The signed distances between minimum and between maximum material solids and between N_1 and N_2 are linked by a simple relationship given in the text.

Figure 6.1.4

Note that these sdist expressions increase with increasing $\Delta\mu$ (though they are not equivalent to $\Delta\mu$ since they include dependence on other degrees of freedom.) Therefore, from C3 it is possible to write the following constraint:

$$\text{sdist}(\Delta\mu,\lambda,N_1,N_2) \geq \text{sdist}(c(\lambda),\lambda,N_1,N_2). \qquad \text{C4}$$

Its right hand side is the signed distance between N_1 and N_2 when the features are in contact. Its left hand side is the signed distance when they are not in contact. It has an advantage over

C3 in that constraints on the expression on its right hand side can be determined easily. These constraints will be derived in the following paragraphs.

The positions of two zones are constrained by the fact that regions containing material do not intersect. Therefore, when the real features are in contact it can be guaranteed that the minimum material regions do not intersect since these are guaranteed to be totally material. Regions containing air, however, do intersect. During contact of the real features the following two conditions hold (see figure 6.1.5). They were also true in the above example.

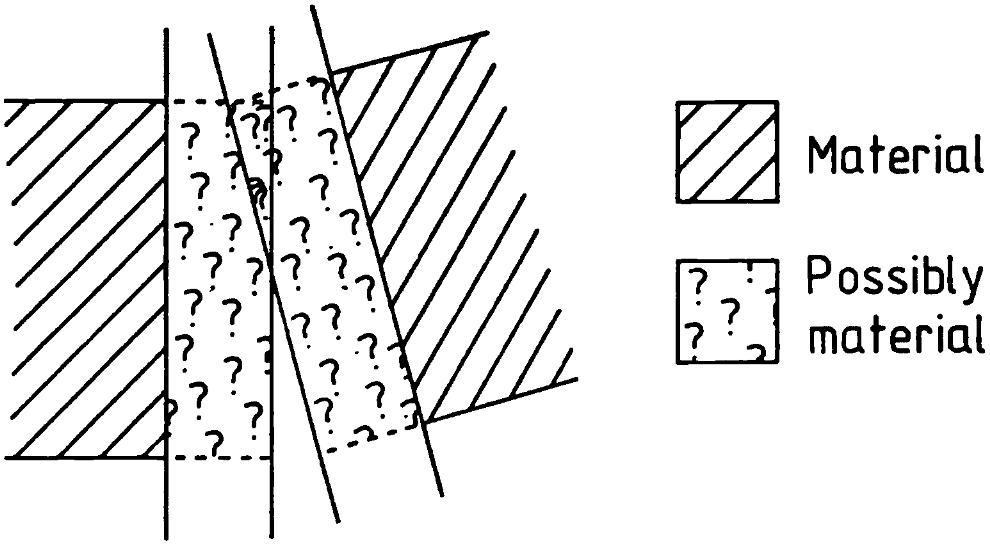
- **Contact Condition 1:** Regions containing only material do not intersect. Therefore the minimum material solids do not intersect. $L_1 \cap L_2 = \phi$.
- **Contact Condition 2:** Regions containing partly air and partly material do intersect. Therefore the maximum material solids do intersect. $M_1 \cap M_2 \neq \phi$.

These conditions only apply strictly to the interiors of the regions involved. So the interiors of L_1 and L_2 do not intersect but the interiors of M_1 and M_2 do intersect. However, it is sometimes convenient and causes no problems to assume that the conditions extend to the boundaries of these regions.

The two contact conditions are converted to an algebraic form using signed distances. The first condition implies that the signed distance between L_1 and L_2 is greater than or equal to zero. Since contact of the real features is occurring $\Delta\mu$ is equal to $c(\lambda)$ giving,

$$\mathrm{sdist}(c(\lambda),\lambda,L_1,L_2) \geq 0. \qquad \text{C5}$$

The second condition implies that the signed distance between M_1 and M_2 is less than or equal to zero and again this occurs at contact,



The "Contact Conditions" imply that the hashed regions (definitely containing material) do not intersect and that the regions containing question marks (possibly containing material) do intersect.

Figure 6.1.5

$$sdist(c(\lambda),\lambda,M_1,M_2) \leq 0. \qquad C6$$

The above two inequalities may be expressed in terms of signed distances between N_1 and N_2 as follows:

$$m_1+m_2 \geq sdist(c(\lambda),\lambda,N_1,N_2) \geq l_1+l_2. \qquad C7$$

The result is a numeric upper and lower bound for the right hand side of C4.

Together with C4 this gives the complete set of constraints on $\Delta\mu$ and λ as,

$$sdist(\Delta\mu,\lambda,L_1,L_2) \geq sdist(c(\lambda),\lambda,N_1,N_2) \quad \text{and} \qquad C8$$
$$m_1+m_2 \geq sdist(c(\lambda),\lambda,N_1,N_2) \geq l_1+l_2.$$

The usefulness of these constraints needs some explanation. Firstly, though, compare them with C1 which concluded the example. On the left hand side of the first inequality is an expression involving DOF-variables. On its right hand side is an expression involving function c (in place of rigid variable c in the example) which is bounded above and below by sums of the tolerance zone offsets in the second inequality.

A more useful form of C8 may be obtained as follows. Make the replacement

$$C(\lambda) = \mathrm{sdist}(c(\lambda), \lambda, N_1, N_2).$$

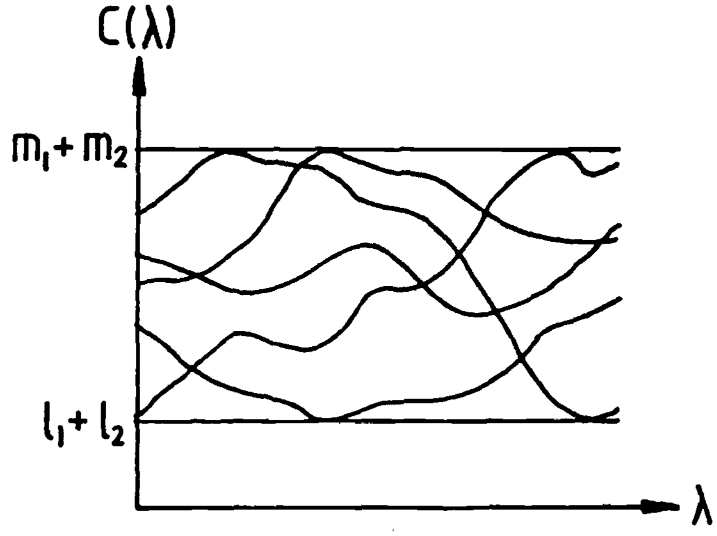
The only important property of $C(\lambda)$ will be its upper and lower bounds given by the second inequality of C8 and so it will never need expansion into more tractable concepts: $C(\lambda)$ can be treated as a single symbol.

The properties of $C(\lambda)$ are illustrated by the graph in figure 6.1.6. The horizontal axis of the graph represents a component of λ and the vertical axis represents $C(\lambda)$. Note that, this graph should not be compared to the one in figure 6.0.1 since their vertical axes refer to different concepts: $C(\lambda)$ is not a "rigid quantity" in that it is variable in any instance of the assembly. Each curve on this graph would arise from a different instance of the contacting surfaces.

The constraints from a relationship between zones containing features in contact can now be written,

$$\mathrm{sdist}(\Delta\mu, \lambda, N_1, N_2) \geq C(\lambda) \quad \text{and} \qquad \text{C9}$$
$$m_1 + m_2 \geq C(\lambda) \geq l_1 + l_2.$$

The first inequality contains a signed distance expression in terms of $\Delta\mu$ and λ. However, it can be replaced by a signed distance expression involving standard DOF-variables by making suitable substitutions. An example of this will be given shortly.



The properties of $C(\lambda)$. Five different $C(\lambda)$ are shown - each corresponds to a different instance of the surfaces involved.

Figure 6.1.6

These constraints can be derived computationally. To appreciate this consider the following three points.

Firstly, the signed distance expression can be derived using the same methods that were used to evaluate signed distances in sections 5.9.3 and 5.9.4. Assume that the features in contact have the same nominal shape (planar, cylindrical etc.). An algorithm for evaluating signed distances is given in appendix 1 and is the same as that used for evaluating signed distances in chapter 5. The only difference is that in chapter 5 the nominal extent of a single feature was used but here the extent of the nominal overlap of two features must be used.

Secondly, the values m_1, m_2, l_1 and l_2 are all obtained directly from the tolerance parameters of the features involved.

Thirdly, $C(\lambda)$ can usually be treated as though it were a variable. The fact that it depends on λ is irrelevant in most situations.

Example

An example will be given to illustrate how constraints are derived in practice in the case of a cylindrical shaft in a cylindrical hole. The hole is subject to a tolerance of unspecified type whose zone is defined as the set difference of two offset-solids with offsets, m_1 and l_1. The tolerance zone of the peg is defined similarly but its tolerance zone has offsets, m_2 and l_2. The type of the tolerance has been left open since it does not affect the form of constraints obtained. The nominal radii of the hole and peg are R and r respectively.

Suitable parameters $\Delta\mu$, $\Delta\lambda_1$, $\Delta\lambda_2$ for a cylindrical peg in a hole were given previously as

$$\Delta\mu = -\sqrt{\Delta y^2+\Delta z^2}$$

$$\Delta\lambda_1 = \mathrm{sign}(\Delta y)\tan^{-1}(\Delta y/\Delta z),\ \Delta\lambda_2=\Delta x.$$

Let C be a function such that

$$m_1+m_2 \geq C(\boldsymbol{\lambda}) \geq l_1+l_2.$$

where $\boldsymbol{\lambda}=(\Delta\lambda_1,\Delta\lambda_2,\Delta\theta,\Delta\phi,\Delta\psi)$. Then, $\Delta\mu$ and $\boldsymbol{\lambda}$ are constrained by a set of constraints of the form C9.

The signed distance in terms of standard degrees of freedom between the nominal peg and hole (with radii R and r) overlapping by an amount E is

$$\sqrt{(\Delta y+(E/2)\Delta\psi)^2 + (\Delta z+(E/2)\Delta\phi)^2} + R-r.$$

This can be substituted for $\mathrm{sdist}(\Delta\mu,\boldsymbol{\lambda},N_1,N_2)$ in C9 to yield,

$$\sqrt{(\Delta y+(E/2)\Delta\psi)^2 + (\Delta z+(E/2)\Delta\phi)^2} + R-r \geq C(\boldsymbol{\lambda})$$

C10

$$m_1+m_2 \geq C(\boldsymbol{\lambda}) \geq l_1+l_2.$$

To show how results can be obtained from these constraints make the simplification that $\Delta z=\Delta\psi=\Delta\phi=0$. Then the first of the above two inequalities becomes,

$$|\Delta y| + R-r \geq C(\lambda)$$

which implies,

$$-C(\lambda) + R-r \geq \Delta y \geq C(\lambda) - (R-r).$$

$C(\lambda)$ can be substituted by any value in the range indicated by the second inequality in C10 (or by any function of DOF-variables which stays within this range). As a result, the maximum and minimum ranges for Δy are found to be,

$$-m_1-m_2 + R-r \geq \Delta y \geq m_1+m_2 - (R-r)$$

and

$$-l_1-l_2 + R-r \geq \Delta y \geq l_1+l_2 - (R-r).$$

Note that for the peg to fit in the hole the upper bound of the minimum range must be greater than the associated lower bound. Therefore,

$$-l_1-l_2 + R-r \geq l_1+l_2 - (R-r)$$

which implies

$$R-r \geq l_1+l_2.$$

Features with more than One Tolerance Type

The example given at the beginning of section 6.1 will be extended to demonstrate what happens when both features have a size tolerance and a position tolerance applied to them. The main purpose of this example is to illustrate a problem which arises when there are two relationships associated with the same potential contact. The situation is shown in figure 6.1.7(i). Again, only one degree of freedom is being considered so the zones are assumed to remain parallel. The two zones of each feature must intersect since the real feature lies in both of them. The network of relationships is shown in figure 6.1.7(ii) and the relationships have been numbered 1 to 4. DOF-variables will be subscripted with the number of the relationship with which they are associated.

Relationships 1 and 2 represent the potential contact of the features. Suppose that the zones in relationship 1 have pairs of offsets m_1, l_1 and m_2, l_2 and that the zones of relationship 2 have pairs of offsets m_3, l_3 and m_4, l_4. Then the constraints in these relationships have the same form as C1:

Relationship 1: $\Delta x_1 \geq c_1, \quad m_1+m_2 \geq c_1 \geq l_1+l_2,$

Relationship 2: $\Delta x_2 \geq c_2, \quad m_3+m_4 \geq c_2 \geq l_3+l_4,$

where the variables c_1 and c_2 are equivalent to the variable c in the first example of this section. They represent the nominal position of the zones when the real features they contain are in contact. Thus, the values of c_1 and c_2 depend on where the real features lie inside the zones.

An important point is that there is a constraint between c_1 and c_2. This is because c_1 and c_2 both depend on the same pair of real features: although relationships 1 and 2 involve different zones they involve the same real features. It will be shown how this dependence can be expressed.

Constraints in relationships 3 and 4 arise from the fact that there must be room in the intersection of the zones for a real feature and can be expressed as, Relationship 3: $\max(m_1-l_3, m_3-l_1) \geq \delta x_3 \geq \min(-m_1+l_3, -m_3+l_1)$

Relationship 4: $\max(m_2-l_4, m_4-l_2) \geq \delta x_4 \geq \min(-m_2+l_4, -m_4+l_2)$

When the real features are in contact then,

$$\Delta x_1 = c_1 \;,\quad \Delta x_2 = c_2 .$$

Δx_1 and Δx_2 represent the relative positions of the real features measured with respect to coordinate systems attached to different zones. These coordinate systems are constrained relative to one another by relationships 3 and 4. The DOF-variables in these relationships are the displacements between the coordinate systems. Therefore,

$$\Delta x_1 + \delta x_4 = \Delta x_2 + \delta x_3 .$$

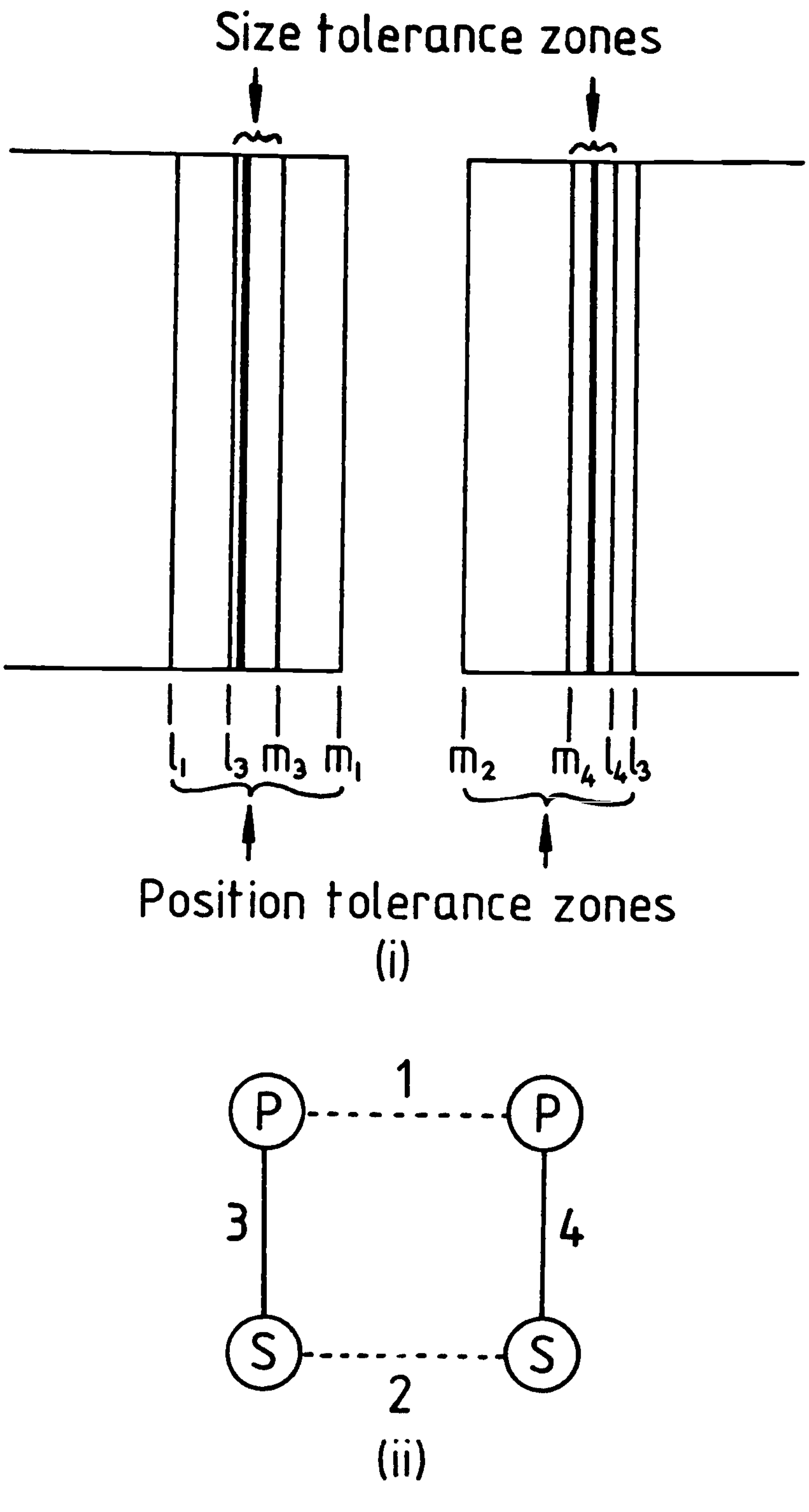
Another way to understand this equality is to notice that the path through the network in figure 6.1.7(ii) consisting of relationships 1 and 4 and the path consisting of relationships 2 and 3 start and end with the same nodes. The sum of the DOF-variables on each path represents the relative position of the start and end nodes and therefore the sums must be identical.

The next step is to suppose that the real features are in contact. Then Δx_1 and Δx_2 can be substituted by c_1 and c_2 respectively to obtain,

$$c_1 + \delta x_4 = c_2 + \delta x_3 . \qquad \text{C11}$$

This constraint can be used to express the fact that it is the same real surfaces that give rise to the constraints in relationships 1 and 2. It is not attached to any relationship but links the variables c_1 and c_2 occurring in relationships 1 and 2.

Another example will be given of features in contact subject to more than one tolerance. Consider a tab in a slot each with a



(i) A potential contact between two features each contained in two zones.
(ii) The resulting network of relationships between the zones.

Figure 6.1.7

position tolerance and a size tolerance and with zone-datum network as figure 6.1.7(ii). DOF-variables Δx_1 and Δx_2 are now bounded above and below by "c-variables" c_1, c_1', c_2 and c_2':

$$c_1' \geq \Delta x_1 \geq c_1 \ , \quad c_2' \geq \Delta x_2 \geq c_2 .$$

There is a constraint relating c_1 and c_1' to c_2 and c_2':

$$c_1'-c_1 = c_2'-c_2 . \qquad \text{C12}$$

This is because Δx_1 and Δx_2 represent displacements between the same real features and so the size of the range of Δx_1 must be the same as the size of the range of Δx_2.

This discussion will now be extended to the more general case where there is more than one degree of freedom. We now have c-functions instead of variables. Constraints between c-functions associated with the same potential contact will be derived. Suppose two such relationships have the following constraints in the form of C9:

$$s_1(\mathbf{x}_1) \geq C_1(\boldsymbol{\lambda}_1),$$

$$a_1 \geq C_1(\boldsymbol{\lambda}_1) \geq b_1$$

and

$$s_2(\mathbf{x}_2) \geq C_2(\boldsymbol{\lambda}_2),$$

$$a_2 \geq C_2(\boldsymbol{\lambda}_2) \geq b_2$$

where $s_1(\mathbf{x}_1)$ and $s_2(\mathbf{x}_2)$ are functions of vectors of standard DOF-variables equivalent to the "$\text{sdist}(\Delta\mu,\lambda,N_1,N_2)$" of C9 and a_1, a_2, b_1 and b_2 are constants.

Also suppose that the vectors of DOF-variables of the relationships between zones belonging to the same feature are $\mathbf{x}_3$ and $\mathbf{x}_4$. Then, a generalised form of C11 is

$$C_1(\boldsymbol{\lambda}_1) + \mathbf{x}_4 = C_2(\boldsymbol{\lambda}_2) + \mathbf{x}_3.$$

Now consider a three-dimensional tab in a three-dimensional slot. There is a constraint with a form similar to C12. Let $\bar{\boldsymbol{\lambda}}=(\Delta\lambda_1,\Delta\lambda_2,1,\Delta\theta,\Delta\phi,\Delta\psi)$ and $\underline{\boldsymbol{\lambda}}=(\Delta\lambda_1,\Delta\lambda_2,-1,\Delta\theta,\Delta\phi,\Delta\psi)$. Then the constraint is

$$C_1(\bar{\boldsymbol{\lambda}}_1) - C_1(\underline{\boldsymbol{\lambda}}_1) = C_2(\bar{\boldsymbol{\lambda}}_2) - C_2(\underline{\boldsymbol{\lambda}}_2).$$

In effect this says that the slop is the same whether it is measured relative to the coordinate systems of one pair of zones or relative to the coordinate systems of the other pair of zones.

-----ooOoo-----

This section has discussed the geometry of contacts between toleranced features. It has been shown how a contact between two features contained in given tolerance zones can be expressed in an algebraic form. The resulting constraints are attached to the relevant relationship between tolerance zones. c-functions were introduced to represent the unpredictable irregularities of the surfaces in contact. In the final subsection it was shown that different c-functions associated with the same potential contact have a constraint between them.

6.2. FORMALISING THE PROBLEM

The constraints obtained in the last section contain c-functions and variables corresponding to sloppy and rigid variations. It is important to understand the semantics of these to see how results can be obtained from the constraints. They do not correspond directly with variations in the real world but rather with variations in **zone-datum structures**.

For this reason, it is helpful to extend the concepts of variational classes and zone-datum structures (ZDS), introduced in section 5.4, to a form convenient for assemblies. A ZDS of a part corresponds to a possible configuration of the zones and datums of the part and can be thought of as a rigid structure. For each ZDS of a part there is a corresponding subset of the variational class of the part whose members are instances of the part with surfaces fitting inside the zones of the ZDS.

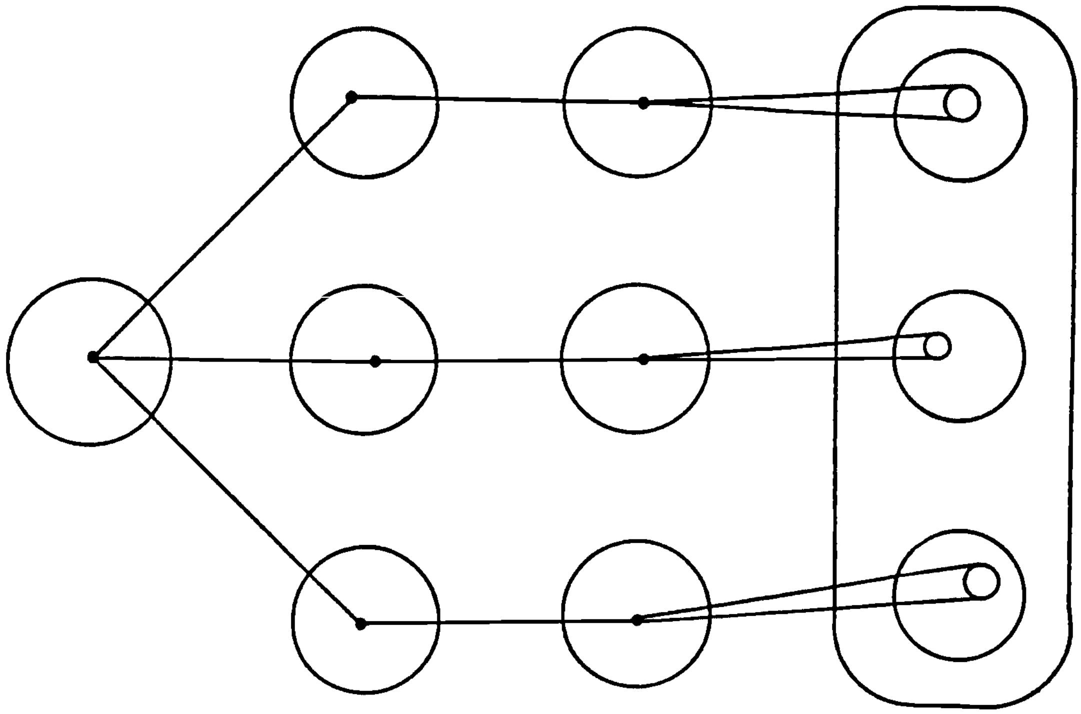
The variational class of an assembly is defined as the cartesian cross-product of the variational classes of the individual parts. The members of this variational class are called instances of the assembly and are n-tuples (where n is the number of parts) each component of which is an instance of one of the parts.

Similarly, the set of ZDS's of an assembly is defined as the cartesian cross-product of the sets of ZDS's of the individual parts. Its members are n-tuples, each component of which is a ZDS of one of the parts, and are called ZDS's of the assembly.

The mapping σ, also introduced in 5.4, between ZDS's and subsets of the variational class, can also be extended to deal with assemblies in the obvious way. If I is the ZDS of an assembly then $\sigma(I)$ is the subset of the variational class of the assembly whose surfaces are contained in the zones of I.

The symbols used for ZDS's and variational classes of assemblies and instances of these will be the same as the symbols for individual parts except that **bold** type will be used.

Figure 6.2.1 summarises these concepts. On the left is a region representing the set of ZDS's of an assembly. Each point in it is associated with a point from each of the ZDS sets of each of the individual parts. A point in one of these regions represents a rigid structure of zones. These points map onto the set of subsets of the variational class of each part. Finally the members of these sets are shown connected to regions in the variational classes all of which are contained in the set of subsets of R^3.



Set of zone-datum structures of an assembly.

Sets of zone-datum structures of individual parts.

Sets of variational classes of individual parts.

Set of all subsets of R^3. Contains sets of subsets of R^3 corresponding to the variational classes of the parts.

Figure 6.2.1

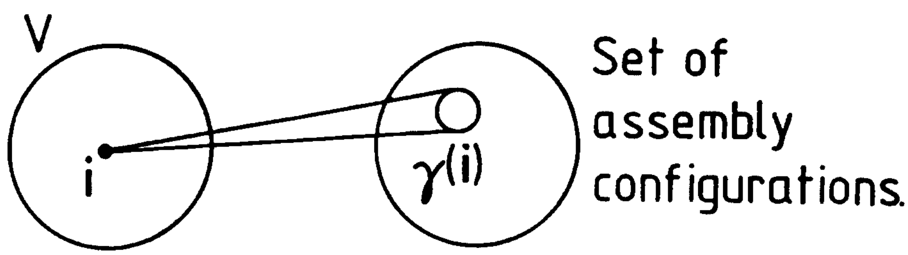
So far the concepts in this section have treated an assembly as a collection of parts without saying anything about the positions of the parts in the assembly. Now, the configuration of an instance of an assembly and of a ZDS of an assembly will be defined.

Formally, the configuration of an assembly can be defined as an n-tuple each component of which is a vector describing the position of one of the parts. Let **c** be an assembly configuration and let C be the space of all configurations. Not all members of C are attainable by an assembly instance since the positions of the parts are constrained by the contacts that occur between them. That is, the parts cannot interfere.

Similarly the configuration of a ZDS of an assembly can be defined. The position of a ZDS of a part is measured with respect to some coordinate system chosen in it. The configuration of a ZDS of an assembly is defined as the n-tuple of positions of the individual ZDS's.

A mapping (figure 6.2.2) exists between the variational class of the assembly V and the set of subsets of configurations of the assembly. Let **i** be a member of the variational class **V** and define $\gamma(\mathbf{i})$ to be the subset of C consisting of configurations for which the parts in **i** do not interfere. The notation $\gamma(\mathbf{S})$, where **S** is a subset of **V**, is used to mean the set,

$$\{\gamma(\mathbf{i}) : \mathbf{i} \in \mathbf{S}\}.$$



The mapping between the variational class of an assembly **V** and the set of assembly configurations.

Figure 6.2.2

Given a ZDS **I** there is a set of subsets of C, $\gamma(\sigma(\mathbf{I}))$, associated with it. Each member of $\gamma(\sigma(\mathbf{I}))$ is a subset of C which can arise from an assembly instance whose surface is contained in the zones of **I**.

The set of sets of configurations that can occur with assembly instances satisfying the tolerance specification is the union of all sets of sets of configurations associated with all ZDS's. It is given by,

$$\bigcup_{\mathbf{I} \in \mathbf{Z}} \gamma(\sigma(\mathbf{I}))$$

where Z is the set of all ZDS's.

It is often useful to deal with the relative positions of two parts or of two features rather than with the configuration of the entire assembly. Although the relative position of a pair of imperfect features is, of course, impossible to define the relative position of two zones that contain the features is easily defined. Given a configuration of a ZDS the relative position of two chosen zones can be isolated. Since there is a set of possible sets of configurations of the ZDS's it follows that there is a set of possible sets of positions of any pair of zones.

In conclusion, the variational class and ZDS of an assembly can be defined. It is possible to define the configuration of a ZDS of an assembly. There is a set of possible sets of configurations which can be taken by a ZDS of an assembly. Correspondingly, there is a set of possible sets of relative positions of any pair of zones in any ZDS of an assembly.

The Interpretation of Variables and c-functions in Terms of Zone-Datum Structures

Now that zone-datum structures of assemblies have been formalised the semantics of rigid and sloppy variables and c-functions can be given.

- Sloppy DOF-variables parameterise the configurations of a ZDS of the assembly. For a given ZDS (with fixed positions for the zones and datums of each part) there is a range of values that can be taken by the sloppy variables corresponding to the possible configurations of the ZDS.
- Rigid variables represent the variations that can occur between different ZDS's. A given ZDS has fixed values of the rigid variables. Constraints on the rigid variables define the set of all possible ZDS's.
- The variability of c-functions represents the variations in the shape of a part that can occur within a ZDS.

6.3. CONSTRAINTS IMPLIED BY PATHS IN THE NETWORK

The introduction of this chapter explained how an assembly of toleranced parts can be represented as a network of zones and datums. Section 6.1 explained how constraints can be associated with the relationships of this network representing potential contacts. This section and the next describe how the constraints can be combined and how results can be obtained from the constraints.

The techniques used are similar to those described in chapters 3 and 5 to analyse a network of constraints. There are three steps:

1. All paths between two chosen nodes are found.
2. The total constraints implied by each path are evaluated.
3. The combined effect of all paths is found.

The presence of both rigid and sloppy variables is an additional complication with toleranced assemblies. As a result the constraints end up with a more complex form.

The resulting constraints can be analysed using the SUPINF algorithm to determine useful information about the assembly. This is the subject of section 6.4.

6.3.1. Finding Paths

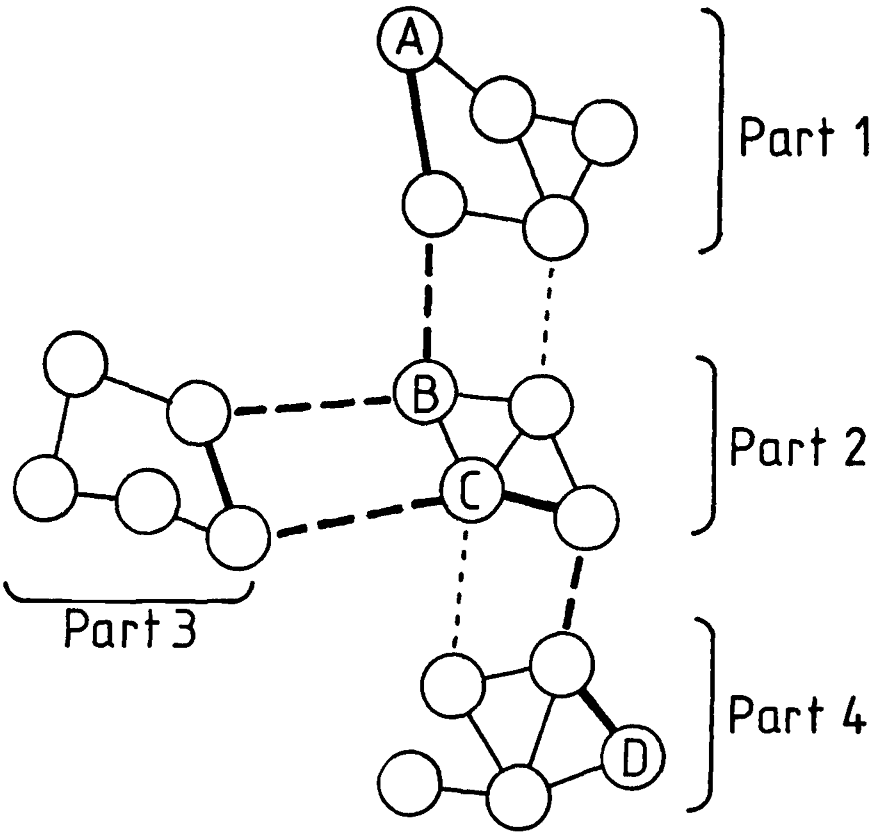
Suppose there are two features, possibly belonging to different parts, whose variation in relative position is to be evaluated. A node associated with each feature is chosen and the constraints are evaluated between these.

The problem of finding all paths between two nodes was dealt with in chapters 3 and 5. In chapter 3 it was suggested that a breadth first search is preferable to a depth first search.

Some of the paths yield redundant constraints and these paths can be ignored. As a result, the quantity of paths to be analysed can be substantially reduced. Two ways that paths may be redundant were the following.

Firstly, from chapter 3, a path may yield constraints which, from a quick estimate, are obviously weaker than constraints implied by paths already investigated.

Secondly, from chapter 5, some paths are found to be redundant by considering the order in which features are manufactured. The relative position of two features is independent of features produced after them. A path is said to be "causally redundant" if it cannot imply constraints for such a reason. Since each part in an assembly is considered to have been manufactured independently of all other parts it follows that causally redundant paths through an assembly can be detected by considering the portions of the path inside each individual part.



A path between nodes A and D is indicated by bold lines. However, rerouting the section of the path between B and C so that it stays within part 2 means that the constraints implied by this section are stronger. Therefore, the entire path as shown can be ignored.

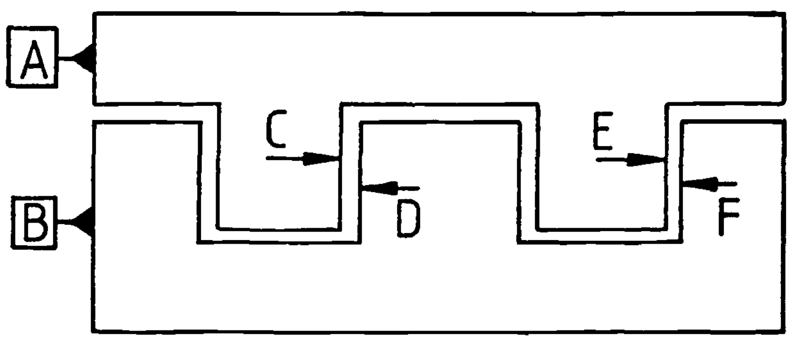
Figure 6.3.1

There is another situation in which certain paths can be rejected: a path which enters and leaves the same part more than once does not imply useful constraints. This can be understood by considering the example of such a path shown in figure 6.3.1. This path can be divided into three sections A to B, B to C and C to D. The constraints implied on A and D by the path are the combination of the constraints in each section. If the constraints on any section can be strengthened by rerouting that section then the entire original path is redundant.

The section between B and C passes through part 3. However, since B and C both belong to the same part, the relative position of B and C is independent of part 3. Therefore, rerouting the section between B and C so that it stays in part 2 produces a path whose constraints are stronger than the original path. As a result, the original path as shown in figure 6.3.1 between A and D can be ignored.

6.3.2. Evaluating Constraints Implied by Individual Paths

Given two nodes associated with the features in the zone-datum network there is a set of sloppy DOF-variables that represent the relative positions of the nodes. Certain paths in the network between the two nodes contribute constraints on these variables.



An assembly in which there are two "paths" between datums A and B.

Figure 6.3.2

An example is given here which shows how constraints could be derived for the relative position of the two datums in the assembly shown in figure 6.3.2. This is the same as the assembly presented in the introduction of this chapter except that the size tolerances have been omitted. A path is shown in figure 6.3.3

between nodes A and B representing the datums. The path goes via C representing one of the tabs and D representing one of the slots. Relationships (1) and (3) are rigid relationships and relationship (2) is a sloppy relationship. The constraints associated with each relationship in the x-direction are shown in the figure. Here capital letters J, K, L, ..., R represent constants which depend on the tolerance parameters. W and w are the widths of the slot and the tab respectively.

Note that relationship (2) provides both an upper and a lower bound for Δx_2 because the tab and slot bound motion in both directions. There is a parameter involved in the constraints which takes on a value ±1. In section 6.1 (p198) the occasional need for this parameter, denoted by $\Delta\lambda_3$, was explained. The symbol $\boldsymbol{\lambda}$ will continue to represent the vector of $\Delta\lambda_1$, $\Delta\lambda_2$ and the rotational DOF-variables. The c-function involved will be written in the format $c(\boldsymbol{\lambda},\pm 1)$ with the second argument representing the value of $\Delta\lambda_3$.

```
         (1)                      (2)                      (3)
A--------------------C---------------------D----------------------B
   J ≤ δx1 ≤ K      Δx2+(W-w)/2 ≥ c(λ,1)       Q ≤ δx3 ≤ R
                       L ≤ c(λ,1) ≤ M
                    -Δx2+(W-w)/2 ≥ c(λ,-1)
                       N ≤ c(λ,-1) ≤ P
```

Relationship (1): $J \leq \delta x_1 \leq K$

Relationship (2): $\Delta x_2 + (W-w)/2 \geq c(\boldsymbol{\lambda},1)$, $L \leq c(\boldsymbol{\lambda},1) \leq M$, $-\Delta x_2 + (W-w)/2 \geq c(\boldsymbol{\lambda},-1)$, $N \leq c(\boldsymbol{\lambda},-1) \leq P$

Relationship (3): $Q \leq \delta x_3 \leq R$

The constraints associated with the path between datums A and D via tab and slot B and C in figure 6.3.2.

Figure 6.3.3

Introduce a DOF-variable Δx which represents the x-component of the relative position of A and B. Δx is equal to the sum of the x-components of the relative positions involved in each of the relationships (1), (2) and (3). Therefore,

$$\Delta x = \delta x_1 + \Delta x_2 + \delta x_3.$$

The upper and lower bounds of Δx in terms of rigid variables and c-functions are found by substituting the upper and lower bounds of Δx_2 to get

$$\begin{gathered} \delta x_1 + c(\lambda,1)-(W-w)/2 + \delta x_3 \\ \leq \Delta x \leq \\ \delta x_1 + c(\lambda,-1)+(W-w)/2 + \delta x_3, \end{gathered} \qquad C1$$

and the rigid variables and c-functions, δx_1, δx_3, $c(\lambda,1)$ and $c(\lambda,-1)$ are constrained by individual relationships as indicated in figure 6.3.3.

6.3.3. Combining Constraints from Multiple Paths

Nodes A and B are also linked by a path which passes via nodes E and F representing the other tab and slot in figure 6.3.1. This path and the DOF-variables constrained by each relationship are shown in figure 6.3.4. The constraints associated with each relationship have not been included because their form is similar to those in figure 6.3.3. The equality

$$\Delta x = \delta x_4 + \Delta x_5 + \delta x_6$$

holds. Substitution of the upper and lower bounds of Δx_5 creates expressions which are upper and lower bounds of Δx. Since these are the only two paths between A and B it follows that these constraints along with C1, provide the total constraints on A and B. Assuming that constraints from different paths are expressed in terms of the same coordinate system it is simply a matter of taking the union of the sets of inequalities obtained from each path.

```
          δx4                    Δx5                    δx6
O--------------------O--------------------O--------------------O
A                    E                    F                    B
```

The path between datums A and B via tab and slot E and F in figure 6.3.2.

Figure 6.3.4

6.3.4. The General Form of Constraints

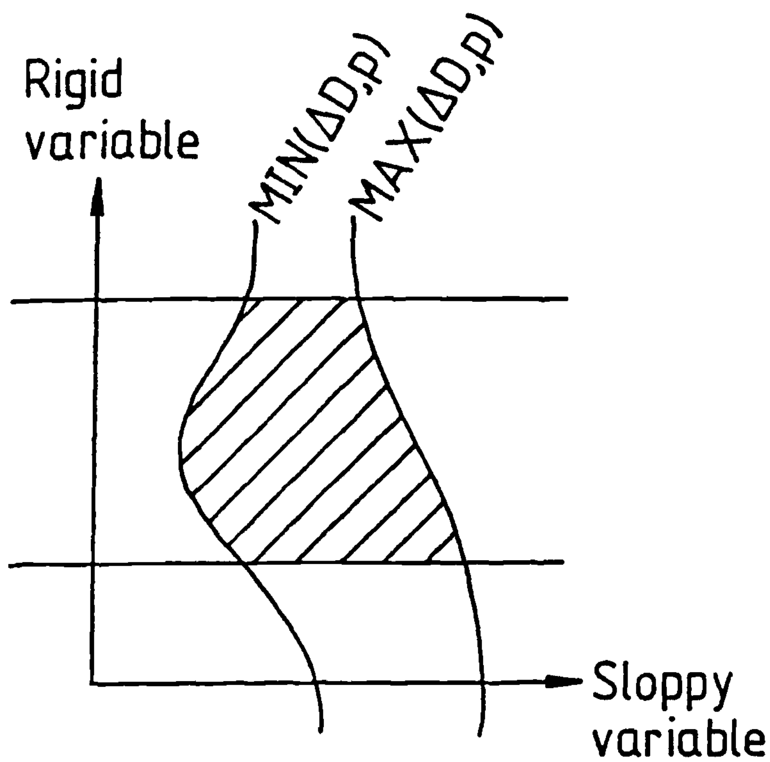
The constraints resulting from the analysis of a path contain three types of item, sloppy variables, rigid variables and c-functions.

- There are at most six sloppy variables and they represent the components of the relative position of the path's terminal nodes.
- There are rigid DOF-variables originating from each rigid relationship in the path and other rigid variables representing the sizes of zones along the path.
- There is a c-function originating from each sloppy relationship.

The set of inequalities (and equalities which can be thought of as two opposing inequalities) can be split into three categories as follows.

- Inequalities involving sloppy DOF-variables. These bound a sloppy DOF-variable above or below by an expression involving rigid variables and c-functions.
- Inequalities arising from individual relationships bounding rigid variables and c-functions.
- Equalities linking the c-functions associated with different relationships but the same potential contact.

Some notation relating to the first of these categories is introduced.



A graph showing MIN(ΔD,p) and MAX(ΔD,p). The horizontal lines represent bounds on a rigid variable. To keep the graph two-dimensional it has been necessary to show only one rigid variable.

Figure 6.3.5

Notation: Let ΔD be a DOF-variable. Define MAX(ΔD,p) and MIN(ΔD,p) to be expressions which contains no sloppy variables and which bound ΔD above and below (respectively) and which are implied by a path denoted p. MAX(ΔD,p) is infinite if the path puts no upper bound on ΔD and MIN(ΔD,p) is negative infinite if the path puts no lower bound on ΔD.

Hence for each path p there are six inequalities, for each of the six sloppy DOF-variables, of the form

$$MIN(\Delta D,p) \leq \Delta D \leq MAX(\Delta D,p).$$

If there are a total of n paths denoted by p_1, p_2, ..., p_n between the two nodes then each sloppy variable ΔD is constrained by

$$\max(MIN(\Delta D,p_1) , \ldots , MIN(\Delta D,p_n)) \leq \Delta D \leq \min(MAX(\Delta D,p_1) , \ldots , MAX(\Delta D,p_n)).$$

The rigid variables and c-functions occurring in these inequalities are constrained by the inequalities in the second and third categories.

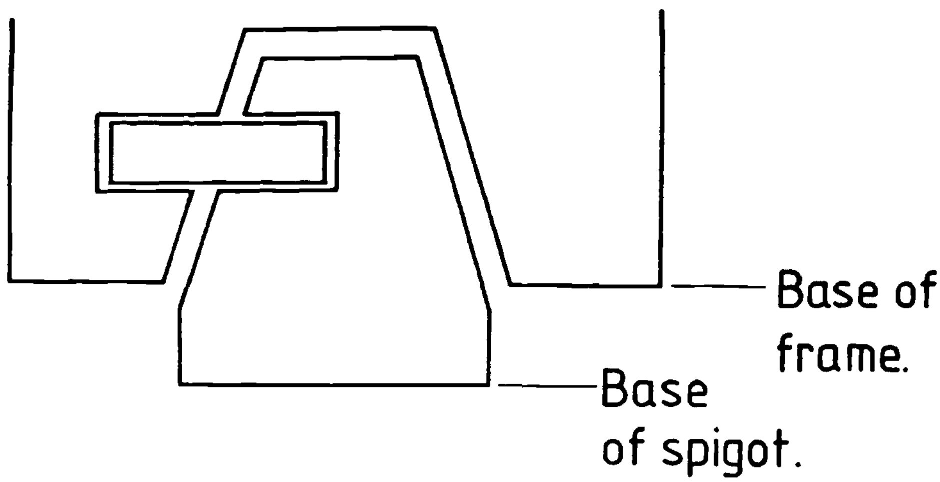
At the beginning of this chapter a simple graph was given which aids visualisation of the solution to these inequalities. It is given again in figure 6.3.5 with the lines representing MAX(ΔD,p) and MIN(ΔD,p) indicated.

6.4. OBTAINING RESULTS FROM THE CONSTRAINTS

This section shows how useful results can be obtained from the constraints obtained in the last section. It is shown how extreme displacements of position or inclination of a part can be determined. It is also shown how extremes of slop can be determined. These results can be used to determine if a toleranced assembly can be guaranteed to satisfy its functionality requirements. For example, it is possible to determine if the parts will ever fail to fit together.

6.4.1. Extreme Positions

An extremal position of a part is the maximum or minimum attainable displacement of the part relative to some other part in a specified direction or rotation. Other parts are assumed to be



A simple assembly.

Figure 6.4.1

free to move and to accommodate to let the displacement of the part increase as much as possible. It is as though a force were applied between two parts so that they move until they are blocked by contacts with other parts.

Consider the assembly shown in figure 6.4.1. The spigot is held in the frame by the peg. Suppose we are interested in the angle that the base of the spigot takes relative to the base of the frame.

The extreme inclination of the base of the spigot relative to the frame could be attained by applying a rotational force to the spigot whilst holding the frame stationary and leaving the peg free to move. The angle at which the spigot comes to rest will vary from one assembly instance to another: there is a range of possible values.

Two questions that might be asked about the extremes of this angle are,

- What is the maximum attainable angle over all instances of the assembly?

- What is the maximum angle that can be guaranteed to be attainable?

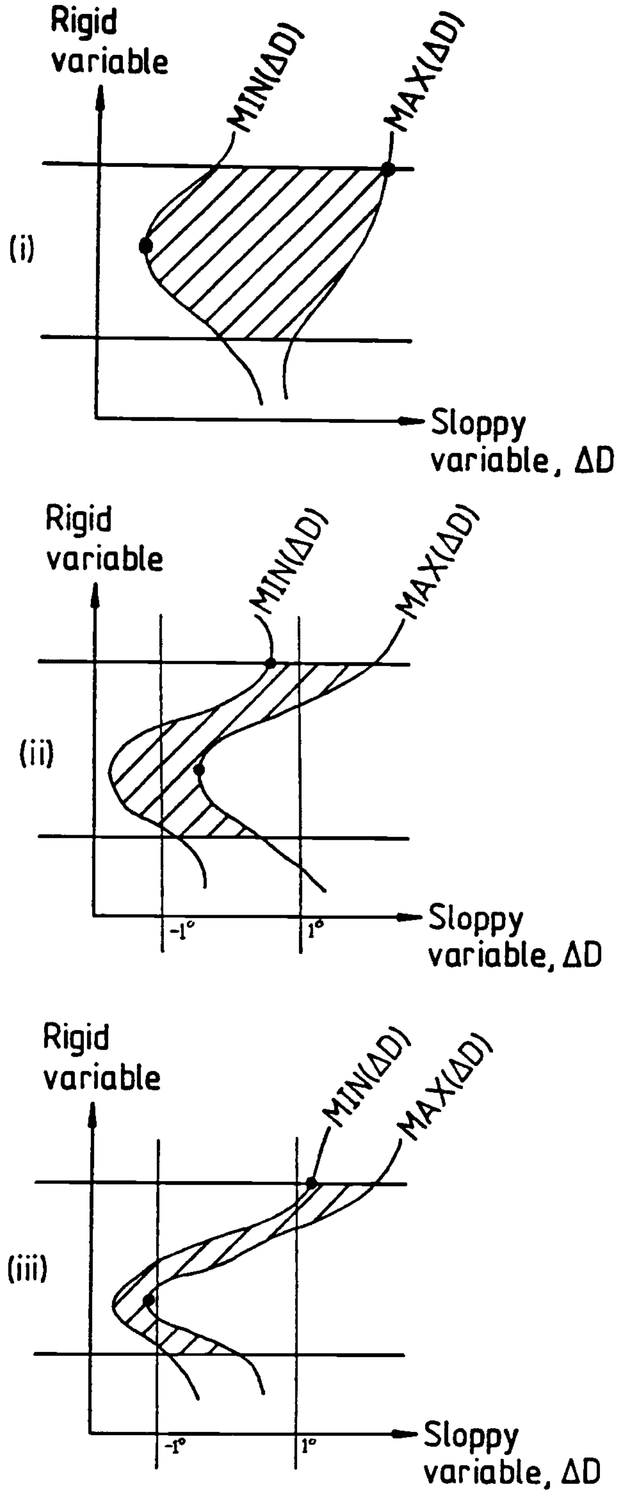
These questions can be represented in a graphical form as shown in figure 6.4.2. The horizontal axis of the graph represents a sloppy DOF-variable associated with the inclination with which we are concerned. The vertical axis is a rigid variable. (This is a simplification, of course, since there are many rigid variables.)

In graph (i) the maximum and minimum values of the angle (represented by DOF-variable ΔD in the graph) that could be attained over all instances of the assembly are indicated. In graphs (ii) and (iii) the minimum value of the upper bound of the angle and the maximum value of its lower bound are illustrated. In graph (ii) an angle in the range -1° to 1° can be guaranteed to be attainable but in (iii) an angle in this range cannot be guaranteed to be attainable. This is because there are allowed values of the rigid variable for which the range of ΔD does not intersect the range -1° to 1°.

The above questions can be answered by finding an expression for the maximum and minimum values of the inclination in terms of rigid variables and c-functions. Then, the extreme values of these expressions can be evaluated.

6.4.1.1. **Evaluating Extreme Positions**

Let the two features whose extreme positions are to be evaluated be associated with nodes A and B. Let the set of inequalities derived from analysis of all paths between A and B be denoted by $I_s \cup I_r$ where I_s contains all inequalities involving sloppy variables and I_r contains all inequalities not involving sloppy variables. Hence I_r includes all inequalities (and equalities) from categories 2 and 3 described at the end of section 6.3.



(i) The two points indicated are the minimum and maximum values of ΔD attainable in any assembly.
(ii) and (iii) The maximum of MIN(ΔD) and the minimum of MAX(ΔD) are indicated.
In (ii) ΔD can attain a value between -1° and 1° for any allowed value of the rigid variable. In (iii) there are allowed values of the rigid variable for which ΔD cannot enter the range -1° to 1°.

Figure 6.4.2

Notation: Let ΔD be the DOF-variable expressing a component of the relative position of A and B. Then MAX(ΔD) is defined to be the minimum of all of MAX(ΔD,p) (introduced in section 6.3.4, p226) for all paths p between A and B. Similarly, MIN(ΔD) is defined to be the maximum of all of MIN(ΔD,p) for all paths p between A and B.

MIN(ΔD) and MAX(ΔD) are expressions in terms of rigid variables and c-functions which bound ΔD above and below. Evaluating these for particular values of the rigid variables and c-functions gives the extreme vlaues of ΔD in an instance of the assembly.

MAX(ΔD) and MIN(ΔD) are represented by the curved lines in the graphs in figure 6.4.2.

MAX(ΔD) and MIN(ΔD) can also be expressed in terms of SUP and INF taken over inequalities I_s. Let R be the set of rigid variables and C be the set of c-functions. Then,

$$\text{MIN}(\Delta D) = \text{INF}(\Delta D,\ I_s,\ R \cup C),$$

$$\text{MAX}(\Delta D) = \text{SUP}(\Delta D,\ I_s,\ R \cup C).$$

The extreme points indicated in figure 6.4.2 can be evaluated by taking the SUP and INF of MIN(ΔD) and MAX(ΔD) over constraints I_r. The points indicated in figure 6.4.2(i) are

$$\text{SUP}(\ \text{MAX}(\Delta D),\ I_r,\ \{\}\ ) \quad \text{and} \quad \text{INF}(\ \text{MIN}(\Delta D),\ I_r,\ \{\}\ ).$$

The points indicated in figures 6.4.2(ii) and (iii) are

$$\text{SUP}(\ \text{MIN}(\Delta D),\ I_r,\ \{\}\ ) \quad \text{and} \quad \text{INF}(\ \text{MAX}(\Delta D),\ I_r,\ \{\}\ ).$$

Suppose that ΔD represents the angle between the base of the spigot and the base of the frame. The requirement that ΔD must be between, say, -3 and 3 degrees in all assemblies can be expressed as

$$-3 \le \text{INF}(\ \text{MIN}(\Delta D),\ I_r,\ \{\}\ )\ ,\ \text{SUP}(\ \text{MAX}(\Delta D),\ I_r,\ \{\}\ ) \le 3.$$

This ensures that the range of ΔD is contained in the range -3 to 3. The requirement that ΔD must be able to attain an angle between, say, -1 and 1 degree in all assemblies can be expressed as

$$-1 \le \text{INF}(\ \text{MAX}(\Delta D),\ I_r,\ \{\}\ )\ ,\ \text{SUP}(\ \text{MIN}(\Delta D),\ I_r,\ \{\}\ ) \le 1.$$

This ensures that the range of ΔD intersects the range -1 to 1. This condition is illustrated by the graph in figure 6.4.2(ii).

6.4.1.2. Evaluation of Extreme Positions in a Simple Example

The example discussed here involves a tab in a slot each satisfying a position tolerance and a size tolerance. This example is simple enough to allow evaluation of extreme positions by simple analysis of the geometry. These results will be compared with an algebraic analysis using SUP and INF as described above.

This example will be used to demonstrate the importance of constraints in category 3 described at the end of section 6.3. Geometric and algebraic analyses will be carried out in the presence and absence of these constraints.

The tolerance zones of the two features, a tab and a slot, are shown in figure 6.4.3. There are four relationships between these zones forming the network which is also shown in figure 6.4.3. Relationships 1 and 2 between zones of different features represent the fact that they contain features in contact. Relationships 3 and 4 between zones of the same feature represent the fact that there must be room in their intersection for a real feature.

To simplify the algebra later in this section both position tolerance zones are given the same offsets, p and -p, and the size tolerance zones are both given offsets s and -s $(p>s>0)$. The nominal separation of the nominal tab and slot is g.

Let us restrict attention to translation in the x-direction. Each relationship has a DOF-variable associated with this direction and they will be denoted by Δx_1, Δx_2, δx_3 and δx_4.

Let $c_1(\boldsymbol{\lambda}_1)$ and $c_2(\boldsymbol{\lambda}_2)$ be c-functions associated with relationships 1 and 2 respectively. Relationships 1 and 2 give rise to inequalities of the form of C9 in section 6.1:

$$s(\Delta\mu_i,\boldsymbol{\lambda}_i) \geq c_i(\boldsymbol{\lambda}_i) \quad , \quad M_i \geq c_i(\boldsymbol{\lambda}_i) \geq L_i \qquad (i=1,2)$$

where, L_i and M_i are constants. Notation used here is similar to that used in C9 section 6.1. In this example,

$$\Delta\mu_i = -|\Delta x_i| \ ,$$

$$\Delta\lambda_1 = \Delta y_i ,\ \Delta\lambda_2 = \Delta z_i ,\ \Delta\lambda_3 = \text{sign}(\Delta x_i) \ \text{ and so}$$

$$\boldsymbol{\lambda}_i = (\Delta y_i, \Delta z_i, \text{sign}(\Delta x_i), \Delta\theta_i, \Delta\phi_i, \Delta\psi_i) \ ,$$

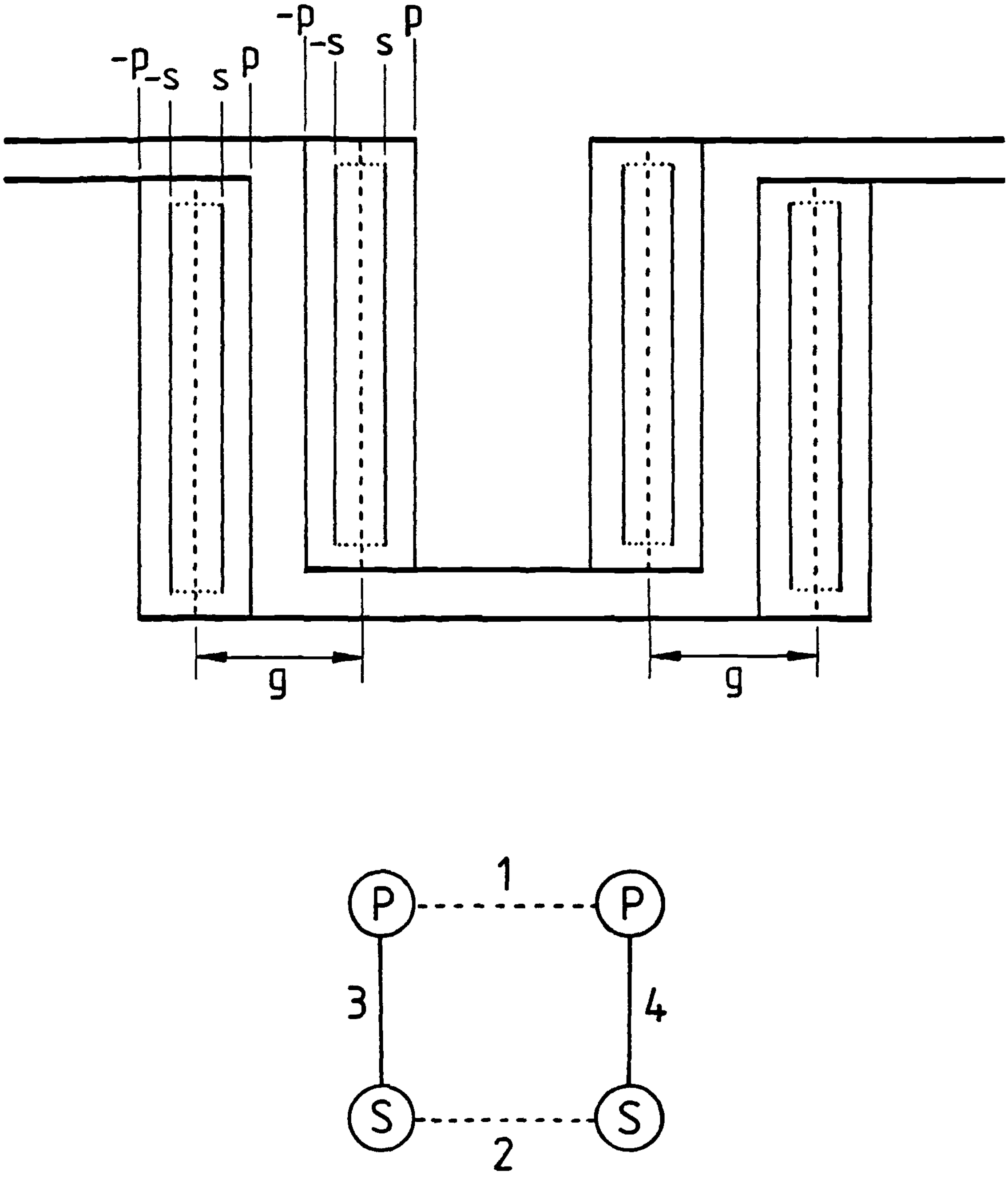
$$s(\Delta\mu_i,\boldsymbol{\lambda}_i) = g+\Delta\mu_i = g-|\Delta x_i| ,$$

$$M_1 = -L_1 = 2p,$$

$$M_2 = -L_2 = 2s.$$

However, since Δx_i are the only degrees of freedom being considered the only important component of $\boldsymbol{\lambda}_i$ is $\text{sign}(\Delta x_i)$. This corresponds to the parameter $\Delta\lambda_3$ introduced on page 198. Therefore, for simplicity, $c_i(\boldsymbol{\lambda}_i)$ will be written $c_i(1)$ or $c_i(-1)$ when the sign of Δx_i is known $(i=1,2)$.

Hence the inequalities from relationships 1 and 2 can be written



Position and size tolerance zones for a tab in a slot. Both position tolerance zones have offsets p and -p. Both size tolerance zones have offsets s and -s. The tab and slot are potentially in contact. The nominal separation of their nominal surfaces is g. The network of relationships numbered 1 to 4 is also shown.

Figure 6.4.3

$$g-|\Delta x_i| \geq c_i(\lambda_i) \quad , \quad M_i \geq c_i(\lambda_i) \geq L_i .$$

The first of these inequalities implies

$$g-c_i(\lambda_i) \geq \Delta x_i \geq c_i(\lambda_i)-g$$

which may be rewritten as

$$g-c_i(1) \geq \Delta x_i \geq c_i(-1)-g.$$

Hence, each relationship implies the constraints as follows.

Relationship 1: $g-c_1(1) \geq \Delta x_1 \geq c_1(-1)-g \quad , \quad 2p \geq c_1(\lambda_1) \geq -2p$

Relationship 2: $g-c_2(1) \geq \Delta x_2 \geq c_2(-1)-g \quad , \quad 2s \geq c_2(\lambda_2) \geq -2s$

Relationship 3: $p+s \geq \delta x_3 \geq -p-s$

Relationship 4: $p+s \geq \delta x_4 \geq -p-s$

Suppose we are interested in finding the extreme relative position of the two position tolerance zones. There are two paths between their associated nodes. The first consists simply of relationship 1 and the second consists of relationships 3, 2 and 4 in that order. Let Δx be the DOF-variable which represents the x-component of the relative positions of these nodes. Δx is related to the other DOF-variables by,

$$\begin{aligned} \Delta x &= \Delta x_1 , \\ \Delta x &= \delta x_3 + \Delta x_2 - \delta x_4 . \end{aligned} \qquad \text{C1}$$

Hence it follows that

$$\Delta x_1 = \delta x_3 + \Delta x_2 - \delta x_4 .$$

When the surfaces of the features are in contact Δx_1 and Δx_2 can be replaced by the c-function expressions that bound them in the constraints for relationships 1 and 2. Either they must both be replaced by their upper bounds or both by their lower bounds depending on which sides of the tab and slot are in contact. After cancelling occurrences of g the following equalities are obtained:

$$c_1(-1) = \delta x_3 + c_2(-1) - \delta x_4, \qquad \text{C2}$$
$$c_1(1) = -\delta x_3 + c_2(1) + \delta x_4.$$

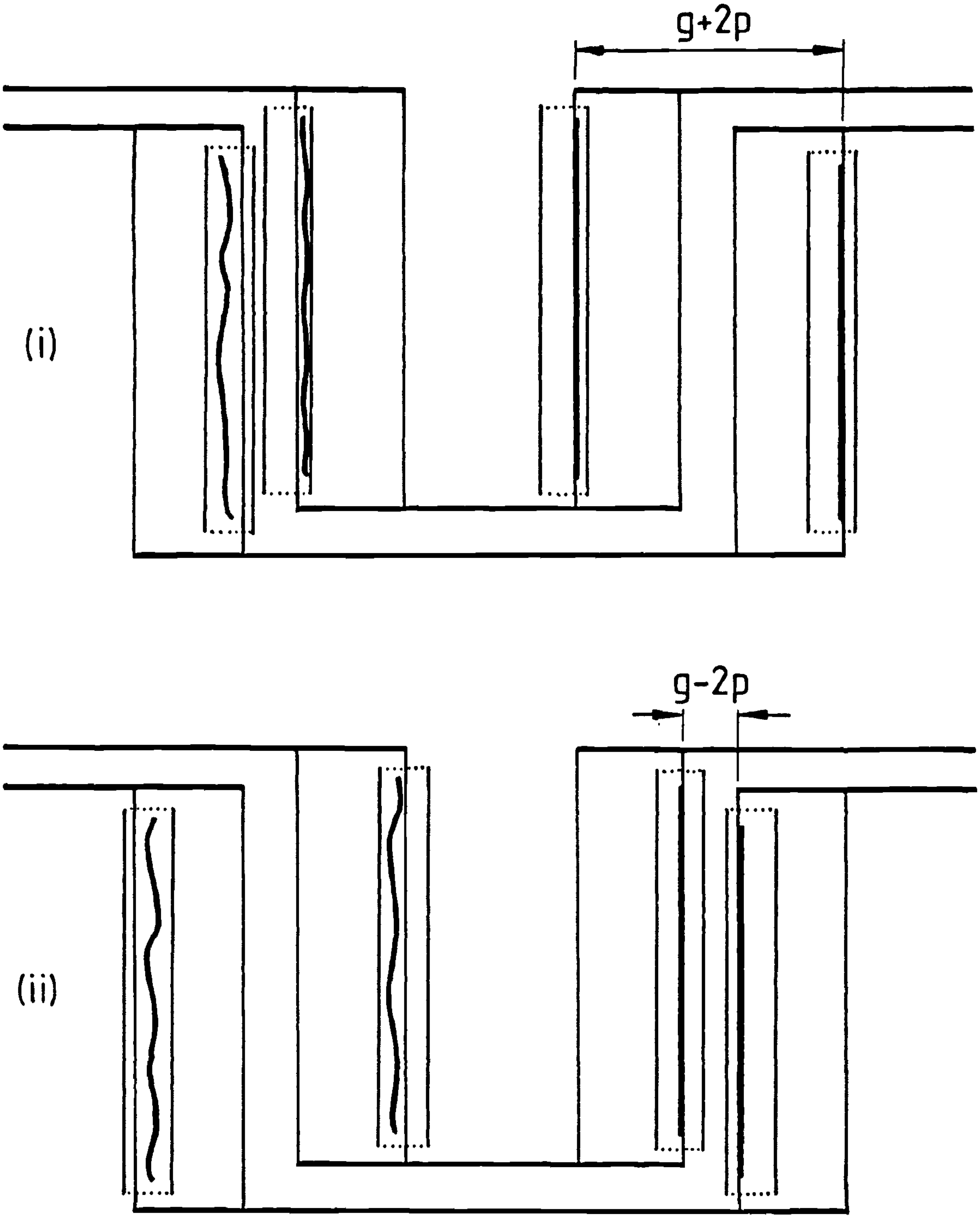
This is the constraint which expresses the fact that relationships 1 and 2 represent contact between the same surfaces. It will be interesting to observe what happens when this constraint is taken into account and what happens when it is ignored. First of all a geometric analysis will be made.

Geometric Analysis

Figure 6.4.4 shows the zones with a real feature lying in their intersection. Thus constraint C2 is satisfied here. Suppose we want to find bounds on $MAX(\Delta x)$, the value of Δx when contact occurs between the surfaces on the right hand side in figure 6.4.4. In figures 6.4.4(i) and (ii) surfaces have been chosen which give maximum and minimum values of $MAX(\Delta x)$. By inspection it can be seen that

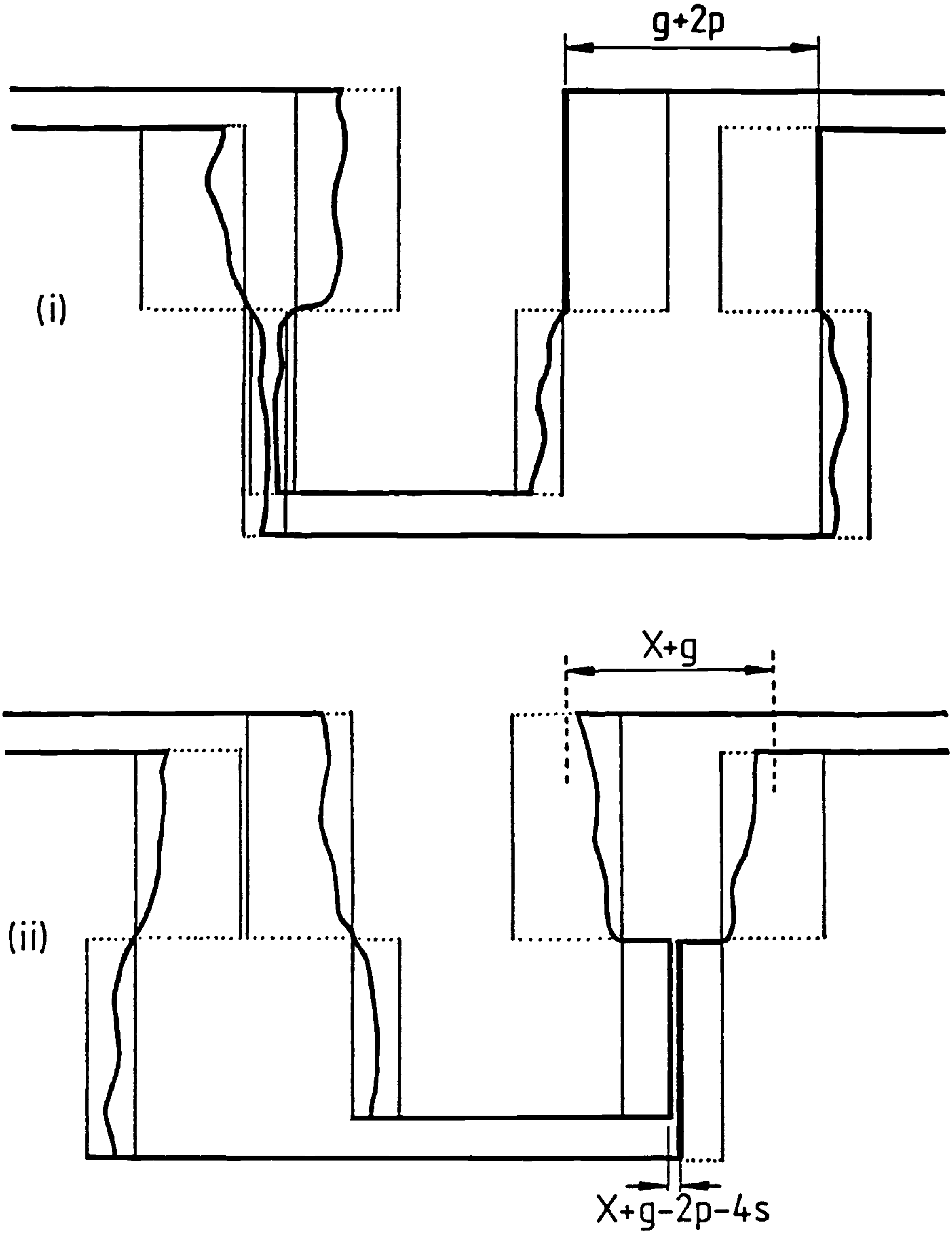
$$g-2p \leq MAX(\Delta x) \leq g+2p.$$

Now suppose that C2 is not satisfied. This means that different surfaces are involved in relationships 1 and 2 and each surface is contained in its respective zone. This situation can be modelled geometrically by considering that each part has two separate parallel features each contained in zones that do not overlap as shown in figure 6.4.5. Although one zone has been displaced in the y-direction from its proper place this does not



The zones of figure 6.4.3 with real surfaces lying in their intersection. The relative positions of the zones and real features in the zones chosen to cause (i) maximum and (ii) minimum of MAX(Δx).

Figure 6.4.4



What would happen if the real feature was not constrained to lie in the intersection of the zones? This hypothetical case is modelled here by separating the zones. Again, relative positions of the zones and real features inside the zones have been chosen to cause (i) maximum and (ii) minimum of MAX(Δx). In (ii) it has been necessary to displace the zones by an amount X in the x-direction since g-2p-4s is negative.

Figure 6.4.5

affect constraints on motion in the x-direction.

The real features in figures 6.4.5(i) and (ii) have been chosen which give rise to extreme values of MAX(Δx). In figure 6.4.5(i) the maximum value of MAX(Δx) is obtained and in figure 6.4.5(ii) its minimum is obtained. By inspection the constraints on MAX(Δx) are seen to be

$$g-(p+2s)-(p+2s) \leq MAX(\Delta x) \leq g+2p.$$

Note that the lower bound of MAX(Δx) is affected by the presence of C2 but the upper bound is unaffected. These results will now be used to demonstrate that correct results can be obtained from an algebraic analysis.

Algebraic Analysis

MAX(Δx), the upper bound of Δx in terms of rigid variables (δx_3 and δx_4) and c-functions ($c_1(\lambda)$ and $c_2(\lambda)$) is obtained from the two equalities in C1 by replacing Δx_1 and Δx_2 by their upper bounds. The minimum is taken of the two results to get

$$MAX(\Delta x) = \min(g-c_1(1),\ \delta x_3+g-c_2(1)-\delta x_4).$$

The SUP and INF can be taken of this expression to find the maximum and minimum possible value of the MAX(Δx). SUP and INF in effect replace occurrences of variables by expressions which are known to bound the variables above or below.

The algorithms SUP and INF are described in detail in Appendix 2. Here, however, it is sufficient to follow the following steps.

1. Expand expressions so that as many terms cancel as possible.

2. Replace c-functions by expressions in terms of rigid variables that bound the c-functions above or below as appropriate. When finding SUP the replacement should cause an increase. When finding INF the replacement should cause a decrease. Evaluating the replacement expression sometimes involves a recursive application of SUP and INF.

3. Cancel as many terms as possible.

4. Repeat step 2 but replacing rigid variables with expressions that bound them above or below. The expressions should not include any rigid variables or c-functions.

5. Simplify.

Note that the cancellations (steps 1 and 3) are necessary in practice to get the best use of SUP and INF. For example applying SUP directly to the degenerate expression x-x (according to rule 5 of appendix 2) without prior simplification would give

SUP(x-x,C,{}) = SUP(x,C,{})-INF(x,C,{})

and this would probably not evaluate to zero. However, prior simplification of the expression would result in an evaluation of SUP(0,C,{}) which is zero.

First let us evaluate SUP and INF of the c-functions so that these expressions are available to make the replacements described in stage 2 above.

The (in)equalities which involve c-functions are obtained from relationships 1 and 2 and from C2. They are repeated here:

1. $2p \geq c_1(\pm 1) \geq -2p$,
2. $2s \geq c_2(\pm 1) \geq -2s$,
3. $c_1(-1) = \delta x_3 + c_2(-1) - \delta x_4$,
4. $c_1(1) = -\delta x_3 + c_2(1) + \delta x_4$.

$c_1(1)$ is bounded above by 2p (from inequality 1) and by $-\delta x_3+c_2(1)+\delta x_4$ (from equality 4). Therefore, $c_1(1)$ is bounded above by $\min[2p,-\delta x_3+c_2(1)+\delta x_4]$. In the notation of appendix 2 this expression is $\text{UPPER}[c_1(1)]$. It can be used to replace $c_1(1)$ but includes $c_2(1)$ which must in turn be replaced by an expression that bounds it above. Upper bounds of $c_2(1)$ are obtained from inequality 2 and from a rearrangement of inequality 4. However, the latter produces an occurrence of $c_1(1)$ which is what we were trying to get rid of in the first case. Hence inequality 4 should be ignored in making this replacement of $c_2(1)$. Therefore $c_2(1)$ is replaced simply by 2s (from inequality 2). The resulting expression that can replace $c_1(1)$ is

$$\text{SUP}[c_1(1),I_r,\{\delta x_3,\delta x_4\}] = \min(2p,-\delta x_3+2s+\delta x_4).$$

The following can be found in a similar way:

$$\text{SUP}[c_2(1),I_r,\{\delta x_3,\delta x_4\}] = \min(\ 2s,\ \delta x_3+2p-\delta x_4),$$

$$\text{INF}[c_1(1),I_r,\{\delta x_3,\delta x_4\}] = \max(-2p,-\delta x_3-2s+\delta x_4),$$

$$\text{INF}[c_2(1),I_r,\{\delta x_3,\delta x_4\}] = \max(-2s,\ \delta x_3-2p-\delta x_4).$$

and

$$\text{SUP}[c_1(-1),I_r,\{\delta x_3,\delta x_4\}] = \min(\ 2p,\ \delta x_3+2s-\delta x_4),$$

$$\text{SUP}[c_2(-1),I_r,\{\delta x_3,\delta x_4\}] = \min(\ 2s,-\delta x_3+2p+\delta x_4),$$

$$\text{INF}[c_1(-1),I_r,\{\delta x_3,\delta x_4\}] = \max(-2p,\ \delta x_3-2s-\delta x_4),$$

$$\text{INF}[c_2(-1),I_r,\{\delta x_3,\delta x_4\}] = \max(-2s,-\delta x_3-2p+\delta x_4).$$

Now let us find the SUP and INF of δx_3 and δx_4 so that these can be inserted at the appropriate point (step 4 above). δx_3 and δx_4 are constrained by the inequalities

$$-p-s \le \delta x_3 \le p+s,$$
$$-p-s \le \delta x_4 \le p+s.$$

Hence

$$SUP(\delta x_3, I_r, \{\}) = p+s \quad , \quad SUP(\delta x_4, I_r, \{\}) = p+s \; ,$$
$$INF(\delta x_3, I_r, \{\}) = -p-s \quad , \quad INF(\delta x_4, I_r, \{\}) = -p-s \; .$$

SUP and INF of MAX(Δx) with C2

First the evaluation of SUP of MAX(Δx) will be demonstrated.

$$SUP(MAX(\Delta x) \; , \; I_r \; , \; \{\} \;)$$
$$= SUP(\min(g-c_1(1), \delta x_3+g-c_2(1)-\delta x_4) \; , \; I_r \; , \; \{\} \;)$$ [see p239]

[Replace $c_1(1)$ and $c_2(1)$ by their INF as given above. Since they are preceded by a minus sign this results in an increase of the expression.]

$$= SUP(\min(g-\max(-2p \; , \; -\delta x_3-2s+\delta x_4 \;) \; , \; g-\max(-2s \; , \; \delta x_3-2p-\delta x_4 \;) +\delta x_3-\delta x_4 \;) \; , \; I_r \; , \; \{\} \;)$$

$$= SUP(\min(2p \; , \; 2s+\delta x_3-\delta x_4 \; , \; 2s+\delta x_3-\delta x_4 \; , \; 2p \;) +g \; , \; I_r, \; \{\} \;)$$

[Replace δx_3 by its SUP, p+s, since it is preceded by a plus sign and δx_4 by its INF, -p-s, since it is preceded by a minus sign.]

$$= \min(\ 2p\ ,\ 2s+p+s+p+s\ ,\ 2s+p+s+p+s\ ,\ 2p\ ) + g$$

$$= \min(\ 2p\ ,\ 2p+4s\ ) + g$$

$$= 2p+g \qquad [\ s>0\ ]$$

Now the evaluation of the INF of $MAX(\Delta x)$ is demonstrated. The expressions replacing $c_1(1)$ and $c_2(1)$ are upper bounds and are respectively,

$$\min(\ 2p\ ,\ -\delta x_3+2s+\delta x_4\ ) \quad \text{and} \quad \min(\ 2s\ ,\ \delta x_3+2p-\delta x_4\ ).$$

These are obtained in the same way as the lower bound replacements described above. Hence,

$$INF(\ MAX(\Delta x),\ I_r\ ,\ \{\}\ )$$

$$= INF(\ \min(g-c_1(1)\ ,\ \delta x_3+g-c_2(1)-\delta x_4)\ ,\ I_r\ ,\ \{\}\ )$$

$$= INF(\ \min(\ g-\min(\ 2p\ ,\ 2s-\delta x_3+\delta x_4\ )\ ,\ g-\min(\ 2s\ ,\ 2p+\delta x_3-\delta x_4\ ) + \delta x_3-\delta x_4\ )\ ,\ I_r\ ,\ \{\}\ )$$

$$= INF(\ \min(\ \max(\ -2p\ ,\ -2s+\delta x_3-\delta x_4\ )\ ,\ \max(\ -2s+\delta x_3-\delta x_4\ ,\ -2p\ )\ ) + g\ ,\ I_r\ ,\ \{\}\ )$$

[Replace δx_3 by $-p-s$ and δx_4 by $p+s$.]

$$= \min(\ \max(\ -2p\ ,\ -p-s-2s-p-s\ )\ ,\ \max(\ -p-s-2s-p-s\ ,\ -2p\ )\ ) + g$$

$$= \min(\ -2p, -2p\ ) + g$$

$$= g-2p.$$

Thus the result is that

$$g-2p \leq MAX(\Delta x) \leq g+2p$$

which is consistent with the geometric analysis illustrated in figure 6.4.4. Hence, it is possible to obtain correct results by using SUP and INF on the constraints obtained from a zone-datum network of an assembly.

SUP and INF of MAX(Δx) without C2

To show that the constraint C2 really is important the evaluations of SUP and INF of MAX(Δx) will be repeated ignoring this constraint.

$c_1(1)$ and $c_2(1)$ are now bounded below simply by -2p and -2s respectively and above by 2p and 2s respectively.

$$SUP(MAX(\Delta x) , I_r , \{\})$$

$$= SUP(min(g-c_1(1) , \delta x_3+g-c_2(1)-\delta x_4) , I_r , \{\})$$

[Step 3 is omitted and steps 2 and 4 performed together. Replace $c_1(1)$, $c_2(1)$, δx_4 by their INFs and δx_3 by its SUP.]

$$= min(g+2p , p+s+g+2s+p+s)$$

$$= g+2p+min(0, 4s)$$

$$= g+2p, \quad \text{since } s>0.$$

$$INF(MAX(\Delta x) , I_r , \{\})$$

$$= INF(min(g-c_1(1) , \delta x_3+g-c_2(1)-\delta x_4) , I_r , \{\})$$

[Replace $c_1(1)$, $c_2(1)$, δx_4 by their SUPs and δx_3 by its INF.]

$= \min(g-2p, -p-s+g-2s-p-s)$

$= -2p+g+\min(0, -4s)$

$= -2p+g-4s$

These are the same results obtained from the geometry shown in figure 6.4.5.

6.4.2. Slop

The slop between two parts in a given direction or rotation is the difference between the upper and lower bounds of the DOF-variable associated with that motion. It is a measure of the looseness of fit of two parts. Basically, the slop is found by taking the difference between the SUP and the INF of the sloppy DOF-variable associated with the motion.

The amount of slop that can be tolerated might be specified in the design specification. An upper bound on slop could be chosen to limit the total amount of slop that could occur and ensure that the parts fitted tightly enough. A suitable lower bound on slop could be chosen to ensure that the parts slide together freely during assembly.

A general requirement is that minimum slop must be greater than zero. If an evaluation of minimum slop ever gives a negative result then this implies that some instances of the parts cannot be assembled. This is because there would be no possible positions of the parts such that they do not interfere. An exception to this rule is that the requirement for an interference fit is equivalent to a requirement for negative slop.

Formally, the slop of one part relative to another can be specified by stating a direction or a rotation along which the slop is to be measured. It is the difference between the two extreme positions that can be associated with this direction. All other parts are assumed to move freely.

In the graphs in figure 6.4.6 slop is taken as the difference between the upper and lower bounds of the sloppy variable within the range of allowed values of the rigid variable. In figure 6.4.6(i) maximum slop is the maximum horizontal width of the shaded region and minimum slop is its minimum width. In figure 6.4.6(ii) the minimum slop is negative: there is a region of the solution to the constraints on the rigid variables for which there is no solution to the sloppy variables. This implies that there are instances of the assembly whose parts do not fit together.

Once a set of inequalities has been obtained as described in section 6.3 evaluation of slop takes place using similar techniques to those described for finding extreme positions. Let the set of inequalities be $I_s \cup I_r$ where I_s includes all inequalities that involve sloppy variables and I_r includes all inequalities that do not involve sloppy variables. Let ΔD be a DOF-variable associated with the direction or rotation along which slop is to be measured. Let R be the set of rigid variables and C be the set of c-functions.

An expression which gives slop in terms of rigid variables and c-functions is

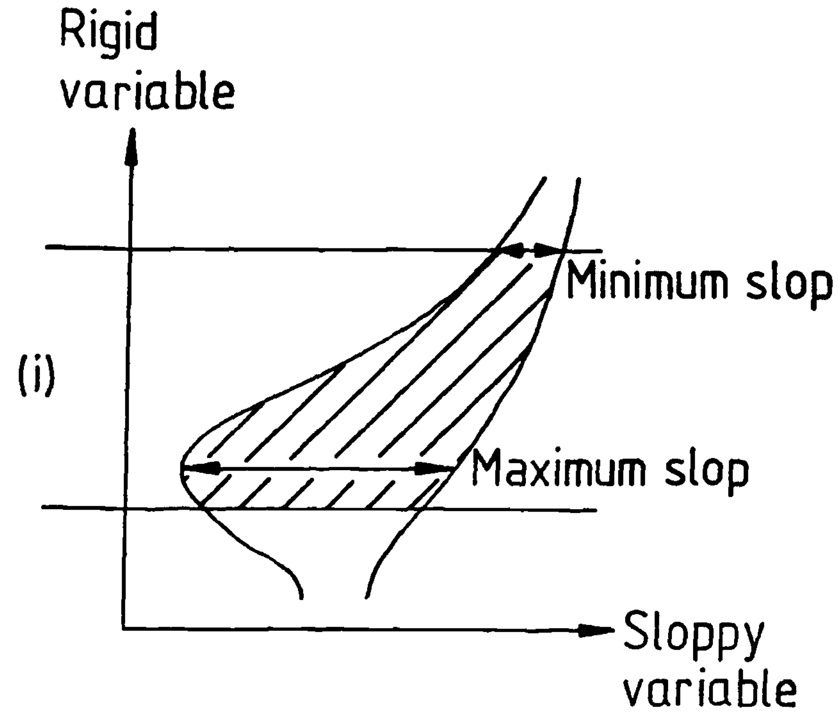
$$SUP(\Delta D,\ I_s,\ R \cup C) - INF(\Delta D,\ I_s,\ R \cup C),$$

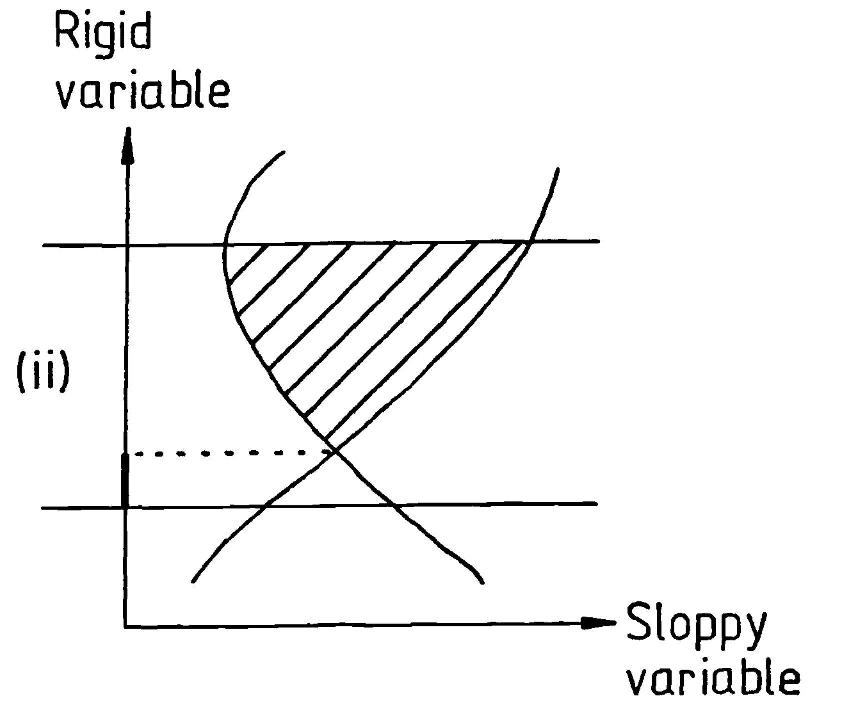
which can be expressed, using the notation defined earlier (p231) as

$$MAX(\Delta D) - MIN(\Delta D).$$

Maximum and minimum slop are obtained simply by taking the SUP and INF, of this expression over I_r:

$$SUP(\ SUP(\Delta D,\ I_s,\ R \cup C) - INF(\Delta D,\ I_s,\ R \cup C)\ ,\ I_r\ ,\ \{\}\ ),$$





(i) Maximum and minimum slop on a graph of a rigid variable against a sloppy variable.
(ii) If minimum slop is negative then there is a solution to the rigid variable which does not have a solution in the sloppy variable. Therefore, some instances of the parts do not fit together.

Figure 6.4.6

$$INF(\; SUP(\Delta D, I_s, R \cup C) - INF(\Delta D, I_s, R \cup C) \; , \; I_r \; , \{\} \;).$$

6.4.2.1. **Evaluation of Slop in a Simple Example**

This example uses the same two features as in the earlier example. The situation is illustrated in figure 6.4.3. As before, it will be interesting to observe what happens when constraints between c-functions are ignored.

Geometric Analysis

First consider figures 6.4.7 (i) and (ii). These show the zones and a real feature lying in their intersection. The real features have been chosen to give maximum and minimum slop. It can be seen by inspection of this figure that

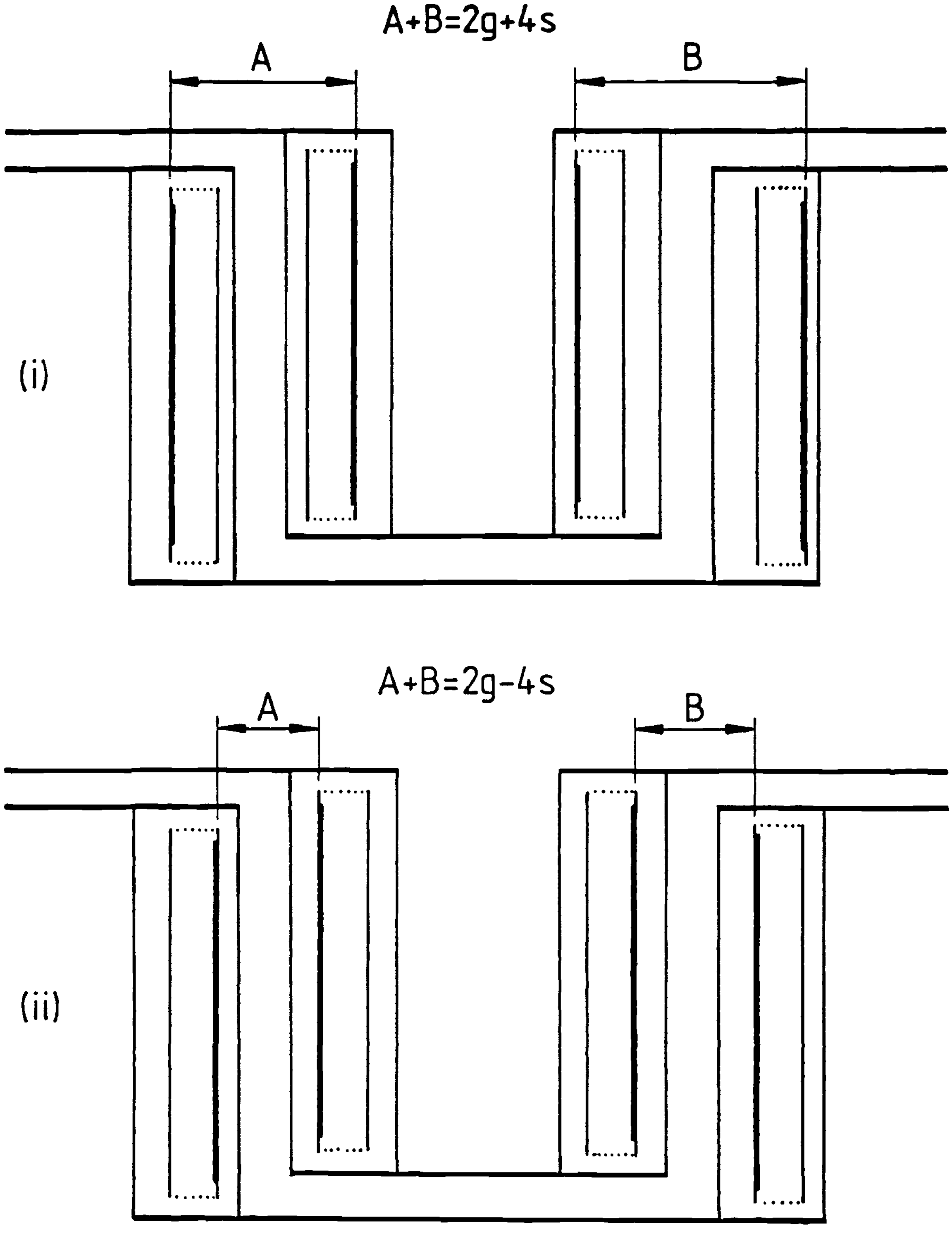
$$2g-4s \leq MAX(\Delta x)-MIN(\Delta x) \leq 2g+4s.$$

Figures 6.4.8 (i) and (ii) model the situation when the constraint, C2, between c-functions is ignored. The bounds on the slop in this situation are given by

$$2g-4s-4p \leq MAX(\Delta x)-MIN(\Delta x) \leq 2g+4s.$$

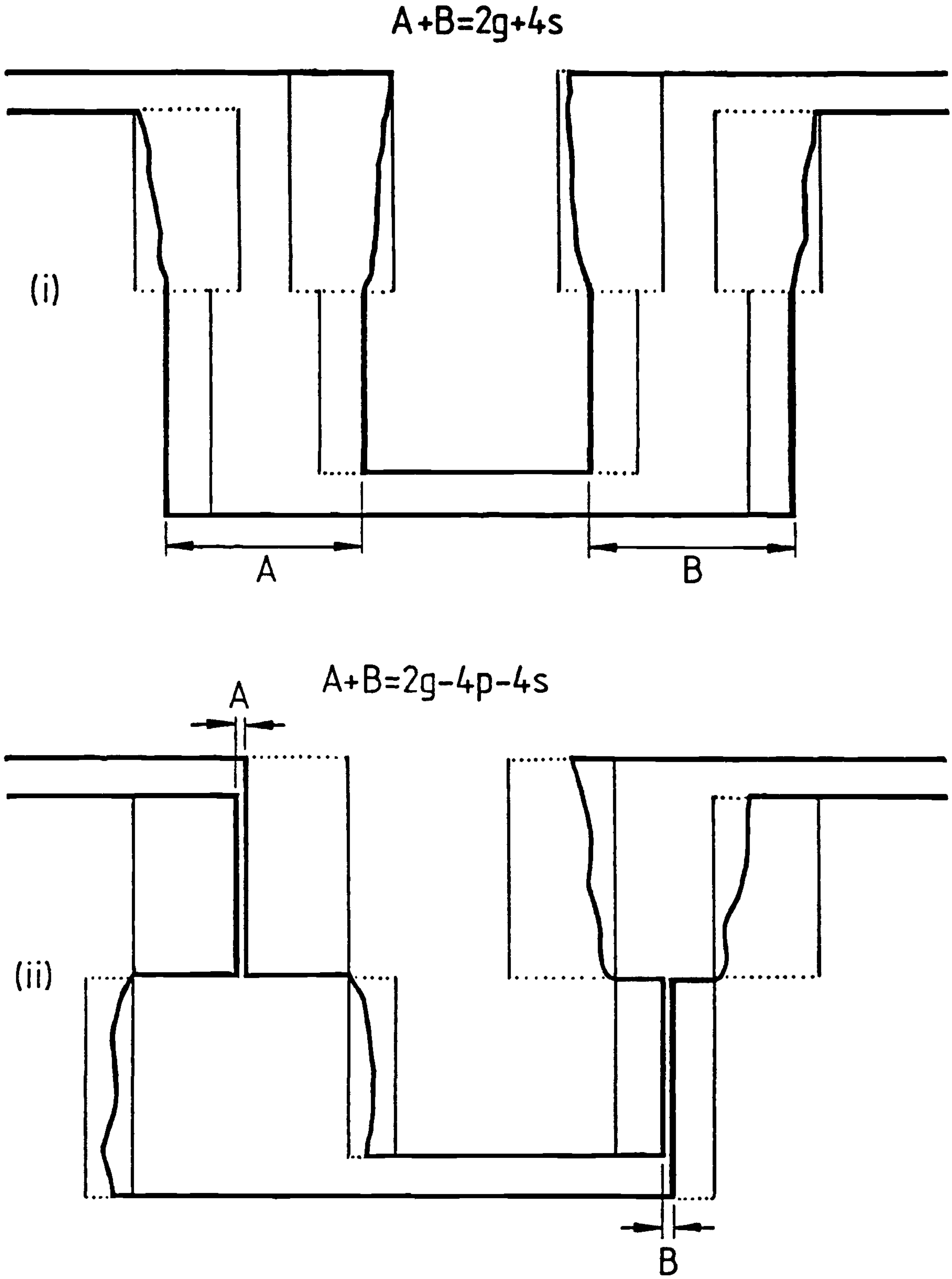
Algebraic Analysis

The evaluation of SUP and INF of MAX(Δx)-MIN(Δx) will be demonstrated in the presence and absence of C2. The five steps for evaluation of SUP and INF given in the previous example (pp239-240) will be followed again here. Recall the following SUPs and INFs of c-functions and rigid variables:



Real features lying in the zones of figure 6.4.3. The relative positions of the zones and the real features have been chosen to give rise to maximum and minimum slop.

Figure 6.4.7



A model of the hypothetical case where the real feature need not lie in the intersection of the zones (cf. figure 6.4.5). The relative positions of the zones and the real features have been chosen to give rise to maximum and minimum slop.

Figure 6.4.8

$$SUP\big(c_1(1),\ I_r,\{\delta x_3,\delta x_4\}\big) = \min(\ 2p,-\delta x_3+2s+\delta x_4),$$

$$SUP\big(c_2(1),\ I_r,\{\delta x_3,\delta x_4\}\big) = \min(\ 2s,\ \delta x_3+2p-\delta x_4),$$

$$INF\big(c_1(1),\ I_r,\{\delta x_3,\delta x_4\}\big) = \max(-2p,-\delta x_3-2s+\delta x_4),$$

$$INF\big(c_2(1),\ I_r,\{\delta x_3,\delta x_4\}\big) = \max(-2s,\ \delta x_3-2p-\delta x_4),$$

$$SUP\big(c_1(-1),I_r,\{\delta x_3,\delta x_4\}\big) = \min(\ 2p,\ \delta x_3+2s-\delta x_4),$$

$$SUP\big(c_2(-1),I_r,\{\delta x_3,\delta x_4\}\big) = \min(\ 2s,-\delta x_3+2p+\delta x_4),$$

$$INF\big(c_1(-1),I_r,\{\delta x_3,\delta x_4\}\big) = \max(-2p,\ \delta x_3-2s-\delta x_4),$$

$$INF\big(c_2(-1),I_r,\{\delta x_3,\delta x_4\}\big) = \max(-2s,-\delta x_3-2p+\delta x_4),$$

and the following SUPs and INFs of rigid variables:

$$SUP(\delta x_3,I_r,\{\}) = p+s \quad , \quad SUP(\delta x_4,I_r,\{\}) = p+s \ ,$$

$$INF(\delta x_3,I_r,\{\}) = -p-s \quad , \quad INF(\delta x_4,I_r,\{\}) = -p-s \ .$$

The DOF-variable Δx represents the x-component of the relative position of the zones of position tolerance. It is related to other DOF-variables (p235) by

$$\Delta x = \Delta x_1 \quad , \quad \Delta x = \delta x_3 + \Delta x_2 - \delta x_4. \qquad (C1)$$

Also recall (p236) the following constraints between c-functions:

$$c_1(-1) = \delta x_3 + c_2(-1) - \delta x_4,$$
$$c_1(1) = -\delta x_3 + c_2(1) + \delta x_4. \qquad (C2)$$

MAX(Δx) and MIN(Δx) are given by

$$MAX(\Delta x) = \min\big(g-c_1(1)\ ,\ \delta x_3+g-c_2(1)-\delta x_4\ \big) \quad \text{and}$$

$$MIN(\Delta x) = \min\big(c_1(-1)-g,\ \delta x_3-g+c_2(-1)-\delta x_4\big)$$

SUP and INF of MAX(Δx)-MIN(Δx) with C2

First assume that C2 holds.

$$\mathrm{SUP}(\ \mathrm{MAX}(\Delta x)-\mathrm{MIN}(\Delta x)\ ,\ I_r\ ,\ \{\}\ )$$

$$= \mathrm{SUP}(\ \min(g-c_1(1),\ \delta x_3+g-c_2(1)-\delta x_4\ ) - \max(c_1(-1)-g,\delta x_3+c_2(-1)-g-\delta x_4)\ ,\ I_r\ ,\ \{\}\ )$$

[Expand to allow some terms to cancel.]

$$\begin{aligned}= \mathrm{SUP}(\ \min(\ & -c_1(1)-c_1(-1)\ ,\\ & -c_1(1)-c_2(-1)-\delta x_3+\delta x_4\ ,\\ & -c_2(1)-c_1(-1)+\delta x_3-\delta x_4\ ,\\ & -c_2(1)-c_2(-1)\ )+2g\ ,\ I_r\ ,\ \{\}\ )\end{aligned}$$

[Replace $c_1(1)$, $c_1(-1)$, $c_2(1)$ and $c_2(-1)$ by their INFs.]

$$\begin{aligned}= \mathrm{SUP}(\ \min(\ & -\max(\ -2p\ ,\ -\delta x_3-2s+\delta x_4\ ) -\\ & \ \max(\ -2p\ ,\ \delta x_3-2s-\delta x_4\ )\ ,\\ & -\max(\ -2p\ ,\ -\delta x_3-2s+\delta x_4\ ) -\\ & \ \max(\ -2s\ ,\ -\delta x_3-2p+\delta x_4\ )-\delta x_3+\delta x_4\ ,\\ & -\max(\ -2s\ ,\ \delta x_3-2p-\delta x_4\ ) -\\ & \ \max(\ -2p\ ,\ \delta x_3-2s-\delta x_4\ )+\delta x_3-\delta x_4\ ,\\ & -\max(\ -2s\ ,\ \delta x_3-2p-\delta x_4\ ) -\\ & \ \max(\ -2s\ ,\ -\delta x_3-2p+\delta x_4\ )\\ )\ & +2g\ ,\ I_r\ ,\ \{\}\ )\end{aligned}$$

$$\begin{aligned}= \mathrm{SUP}(\ \min(\ & \min(\ 4p\ ,\ 2p+2s-\delta x_3+\delta x_4\ ,\ 2s+2p+\delta x_3-\delta x_4\ ,\ 4s\ )\ ,\\ & \min(\ 2p+2s-\delta x_3+\delta x_4\ ,\ 4p\ ,\ 4s\ ,\ 2s+2p+\delta x_3-\delta x_4\ )\ ,\\ & \min(\ 2s+2p+\delta x_3-\delta x_4\ ,\ 4s\ ,\ 4p\ ,\ 2p+2s-\delta x_3+\delta x_4\ )\ ,\\ & \min(\ 4s\ ,\ 2s+2p+\delta x_3-\delta x_4\ ,\ 2p+2s-\delta x_3+\delta x_4\ ,\ 4p\ )\\ )\ & +2g\ ,\ I_r\ ,\ \{\}\ )\end{aligned}$$

$$= \text{SUP}\big(\min(4p\ ,\ 4s\ ,\ 2p+2s+\delta x_3-\delta x_4\ ,\ 2p+2s-\delta x_3+\delta x_4\ ) +2g\ ,\ I_r\ ,\ \{\}\ \big)$$

[Replace δx_3 and δx_4 by their SUPs when they are preceded by a plus sign and by their INFs when they are preceded by a minus sign.]

$$= \min(4p\ ,\ 4s\ ,\ 2p+2s+2(p+s)\ ,\ 2p+2s+2(p+s)\ ) +2g$$

$$= \min(4p\ ,\ 4s\ ,\ 4s+4p\ ) +2g \qquad [\ p>s>0\ ]$$

$$= 4s+2g$$

Now the minimum slop will be calculated.

$$\text{INF}\big(\text{MAX}(\Delta x)-\text{MIN}(\Delta x)\ ,\ I_r\ ,\ \{\}\ \big)$$

$$= \text{INF}\big(\min(g-c_1(1)\ ,\ \delta x_3+g-c_2(1)-\delta x_4\ ) - \max(c_1(-1)-g\ ,\ \delta x_3+c_2(-1)-g-\delta x_4)\ ,\ I_r\ ,\ \{\}\ \big)$$

[Expand to allow some terms to cancel.]

$$= \text{INF}\big(\min(-c_1(1)-c_1(-1)\ ,\ -c_1(1)-c_2(-1)-\delta x_3+\delta x_4\ ,\ -c_2(1)-c_1(-1)+\delta x_3-\delta x_4\ ,\ -c_2(1)-c_2(-1)\ ) +2g\ ,\ I_r\ ,\ \{\}\ \big)$$

[Replace $c_1(1)$, $c_1(-1)$, $c_2(1)$ and $c_2(-1)$ by their SUPs.]

$$= \text{INF}\big(\min(-\min(2p\ ,\ -\delta x_3+2s+\delta x_4\ ) - \min(2p\ ,\ \delta x_3+2s-\delta x_4\ )\ ,$$

$$-\min(2p\ ,\ -\delta x_3+2s+\delta x_4\ ) - \min(2s\ ,\ -\delta x_3+2p+\delta x_4\ ) -\delta x_3+\delta x_4\ ,$$

$$-\min(2s\ ,\ \delta x_3+2p-\delta x_4\ ) - \min(2p\ ,\ \delta x_3+2s-\delta x_4\ ) +\delta x_3-\delta x_4\ ,$$

$$-\min(\ 2s\ ,\ \delta x_3+2p-\delta x_4\ )\ -$$
$$\min(\ 2s\ ,\ -\delta x_3+2p+\delta x_4\ )$$
$$)\ +2g\ ,\ I_r\ ,\ \{\}\ )$$

$$= \text{INF}(\ \min(\ \max(\ -4p\ ,\ -2p-2s-\delta x_3+\delta x_4\ ,\ -2s-2p+\delta x_3-\delta x_4\ ,\ -4s\ )\ ,$$
$$\max(\ -2p-2s-\delta x_3+\delta x_4\ ,\ -4p\ ,\ -4s\ ,\ -2s-2p-\delta x_3+\delta x_4\ )\ ,$$
$$\max(\ -2s-2p+\delta x_3-\delta x_4\ ,\ -4s\ ,\ -4p\ ,\ -2p-2s+\delta x_3-\delta x_4\ )\ ,$$
$$\max(\ -4s\ ,\ -2s-2p+\delta x_3-\delta x_4\ ,\ -2p-2s-\delta x_3+\delta x_4\ ,\ -4p\ )$$
$$)\ +2g\ ,\ I_r\ ,\ \{\}\ )$$

[Replace δx_3 and δx_4 by their SUPs when they are preceded by a plus sign and by their INFs when they are preceded by a minus sign.]

$$= \min(\ \max(\ -4p\ ,\ -2p-2s-2(p+s)\ ,\ -2p-2s-2(p+s)\ ,\ -4s\ )\ ,$$
$$\max(\ -2p-2s-2(p+s)\ ,\ -4p\ ,\ -4s\ ,\ -2s-2p-2(p+s)\ )\ ,$$
$$\max(\ -2s-2p-2(p+s)\ ,\ -4s\ ,\ -4p\ ,\ -2p-2s-2(p+s)\ )\ ,$$
$$\max(\ -4s\ ,\ -2s-2p-2(p+s)\ ,\ -2p-2s-2(p+s)\ ,\ -4p\ )$$
$$)\ +2g\ ,\ I_r\ ,\ \{\}\ )$$

$$= \min(\ \max(\ -4p\ ,\ -4p-4s\ ,\ -4p-4s\ ,\ -4s\ )\ ,$$
$$\max(\ -4p-4s\ ,\ -4p\ ,\ -4s\ ,\ -4s-4p\ )\ ,$$
$$\max(\ -4s-4p\ ,\ -4s\ ,\ -4p\ ,\ -4p-4s\ )\ ,$$
$$\max(\ -4s\ ,\ -4s-4p\ ,\ -4p-4s\ ,\ -4p\ )$$
$$)\ +2g\ ,\ I_r\ ,\ \{\}\ )$$

$$= \min(\ -4s\ ,\ -4s\ ,\ -4s\ ,\ -4s\ )\ +2g \quad [\ p>s>0\ ]$$

$$= -4s+2g.$$

These examples have demonstrated that it is possible to use the SUPINF algorithm to derive maximum and minimum slop from the constraints contained in a zone-datum network. The results are the same as those obtained from a geometric analysis.

SUP and INF of MAX(Δx)-MIN(Δx) without C2

The effect of ignoring the constraints C2 between c-functions associated with the same potential contact was modelled geometrically. Now it will be shown that repeating the above evaluations of SUP and INF of MAX(Δx)-MIN(Δx) whilst ignoring the constraint between c-functions yields the result expected from geometry.

$$\mathrm{SUP}\big(\ \mathrm{MAX}(\Delta x)-\mathrm{MIN}(\Delta x)\ ,\ I_r\ ,\ \{\}\ \big)$$

$$= \mathrm{SUP}\big(\ \min(g-c_1(1)\ ,\ \delta x_3+g-c_2(1)-\delta x_4\ )$$
$$\max(c_1(-1)-g,\ \delta x_3+c_2(-1)-g-\delta x_4)\ \ ,\ I_r\ ,\ \{\}\ \big)$$

[Expand to allow some terms to cancel.]

$$= \mathrm{SUP}\big(\ \min\big(\ -c_1(1)-c_1(-1)\ ,$$
$$-c_1(1)-c_2(-1)-\delta x_3+\delta x_4\ ,$$
$$-c_2(1)-c_1(-1)+\delta x_3-\delta x_4\ ,$$
$$-c_2(1)-c_2(-1)\ \big)\ +2g\ \ ,\ I_r\ ,\ \{\}\ \big)$$

[The INFs of $c_1(\pm1)$ and $c_2(\pm1)$ are now simply -2p and -2s respectively.]

$$= \mathrm{SUP}\big(\ \min\big(\ 2p-(-2p)\ ,$$
$$2p-(-2s)-\delta x_3+\delta x_4\ ,$$
$$2s-(-2p)+\delta x_3-\delta x_4\ ,$$
$$2s-(-2s)\ \ \big)\ +2g\ ,\ \ I_r\ ,\ \{\}\ \big)$$

[Replace $\pm\delta x_3$ and $\pm\delta x_4$ by p+s.]

$$= \min\big(\ 4p\ ,\ 4s\ ,\ 2p+2s+2(p+s)\ ,\ 2p+2s+2(p+s)\ \big)\ +2g$$

$$= \min\big(\ 4p\ ,\ 4s\ ,\ 4s+4p\ )\ +2g \qquad [\ p>s>0\ ]$$

$$= 4s+2g$$

And the minimum slop ...

$$\mathrm{INF}(\ \mathrm{MAX}(\Delta x)-\mathrm{MIN}(\Delta x)\ ,\ I_r\ ,\ \{\}\ )$$

$$= \mathrm{INF}(\ \min(g-c_1(1),\ \delta x_3+g-c_2(1)-\delta x_4\ ) - \max(c_1(-1)-g, \delta x_3+c_2(-1)-g-\delta x_4)\ ,\ I_r\ ,\ \{\}\ )$$

[Expand to allow some terms to cancel.]

$$= \mathrm{INF}(\ \min(\ -c_1(1)-c_1(-1)\ ,\ -c_1(1)-c_2(-1)-\delta x_3+\delta x_4\ ,\ -c_2(1)-c_1(-1)+\delta x_3-\delta x_4\ ,\ -c_2(1)-c_2(-1)\ )\ +2g\ ,\ I_r\ ,\ \{\}\ )$$

[The SUPs of $c_1(\pm1)$ and $c_2(\pm1)$ are 2p and 2s respectively.]

$$= \mathrm{INF}(\ \min(\ -2p-2p\ ,\ -2p-2s-\delta x_3+\delta x_4\ ,\ -2s-2p+\delta x_3-\delta x_4\ ,\ -2s-2s\ )\ +2g\ ,\ I_r\ ,\ \{\}\ )$$

[Replace $\pm\delta x_3$ and $\pm\delta x_4$ by p+s.]

$$= \min(\ -4p\ ,\ -2p-2s-2(p+s)\ ,\ -2s-2p-2(p+s)\ ,\ -4s\ )\ +2g$$

$$= -4p-4s+2g$$

This is the value of minimum slop predicted from the situation shown in figure 6.4.5(ii).

It follows that the SUPINF algorithm can be used to evaluate properties of an assembly including extreme attainable positions and maximum and minimum slop. From this information the satisfaction of design requirements by an assembly to which tolerances have been allocated can be decided.

6.5. CONCLUSION

This chapter has shown that an assembly of parts can be represented as a zone-datum network. Relationships have been introduced between tolerance zones which contain features that are potentially in contact. If the features have more than one tolerance zone then there will be several relationships to represent each potential contact. Since each of these relationships involves the same pair of real surfaces a constraint must be introduced to represent this fact.

The geometry of imperfect features potentially in contact was discussed in detail. The resulting algebraic constraints contain "c-functions" which express the effect of the imperfections of the surfaces. Although, the exact nature of c-functions are not known it can be shown that they are bounded above and below in a simple way. After this it is possible to treat them in the same way as a variable bounded above and below.

In practice the constraints associated with relationships between zones of different features can be evaluated using signed distances. This technique was also used in chapter 5.

The network has a similar structure to networks dealt with in earlier chapters and so can be handled in a similar way. The form of the constraints obtained is complicated by the presence of two types of variable, rigid variables and sloppy variables. Results can be obtained from the constraints using the SUPINF algorithm.

The tools required for analysis of assemblies of parts, including signed distances, path finding and the SUPINF algorithm were all discussed in previous chapters.

Chapter 7: **CONCLUSION**

This thesis has tackled the problem of representation and analysis of geometric tolerances in assemblies of parts. A computational representation of geometrically toleranced parts has been achieved. Assemblies of such parts can also be represented and this has been supported by an understanding of the geometry of contacts between toleranced features.

The representation of geometric tolerances is based on the sound formalism described by Requicha (1983a). This makes use of tolerance zones with properties depending on the type of tolerance and datums which are points, lines or planes fixed in the part.

Basically, parts were represented as a network of relationships between tolerance zones and datums. Constraints between zones and datums are derived from knowledge of the geometry and are attached to the arcs of the network. In some types of relationships there are only a few physical possibilities and a catalogue could be constructed of constraints for each of these. In other types of relationship the constraints are derived from standard procedures that have been described in detail.

An assembly of parts can be analysed to find the how the positions of its parts or their features are constrained. This is done by combining the constraints in the network to find the total constraints between two of its nodes. These constraints express the possible variation in position of the tolerance zones or datums associated with the nodes. Given two features of a part it is possible to calculate the variation in their position caused by the tolerance specification. Given two features of different parts in an assembly it is possible to find the variation in their relative position taking into account variations in size and shape of the parts and sloppy fits between parts in the assembly. It is possible to calculate whether the parts will ever fail to be able to fit together.

These results allow the functionality requirements of a part or of an assembly of parts to be checked.

The following are some aspects of the work which, as far as it is known, have not been covered by previous work.

- Geometric tolerancing has been dealt with in terms of representation and inferencing. Geometric tolerances offer the advantage of increased realism over dimensional tolerances since the latter assume that the form of a surface is perfect.
- Methods for representing and analysing assemblies of toleranced parts have been produced.
- The geometry of contacts between toleranced features has been formalised.
- There is no restriction on the regions of surface to which a tolerance can be applied except that the region must be definable as a feature (simple of composite). This means that parts in an assembly can have incompatible tolerance allocations.
- A classification of geometric relationships between tolerance zones and datums has been produced.
- Parts and assemblies of parts are represented as a network of tolerance zones and datums. No explicit mention is made of their features.

Some major assumptions of the work are the following.

- All uncertainties are assumed to be small. That is, variations in size and position are small compared to the nominal size of features.
- Parts are assumed to be rigid and do not change their shape at all.
- Parts in an assembly are not held together in any way

(for example by bolts or by glue). Positions of parts are constrained only by the contacts between them.

- Parts are not subject to any forces such as gravity. Hence they are free to move in any direction.
- The statistics of tolerances has not been dealt with. This is equivalent to making the assumption that a toleranced distance between two features can have any value within its given range with equal probability.
- Tolerances on different parts or on different features of the same part are assumed to be completely independent.
- Parts are chosen for assembly in a random manner.

7.1. The Basic Approach

Throughout, networks are built of constraints on the relative positions of items which are either nominal parts or tolerance zones and datums. The constraints are represented as inequalities derived from consideration of geometric relationships between the items. The combined effect of the constraints is deduced by making inferences on the constraints.

In chapter 3 the problem was to find the uncertainty in position of parts in an assembly. The parts were assumed to be perfectly formed but poorly fitting. A network was built with nodes representing parts and arcs representing possible contacts between them. The constraints derive from the fact that features of the parts do not interfere. It was assumed that the parts have fixed and known dimensions. It was shown how the amount of slop of parts in an assembly can be calculated.

Chapter 4 described a formalism of geometric tolerances developed by Requicha. The constraints that exist between tolerance parameters associated with the same feature were investigated. Some additional comments were made about tolerance schemes and toleranced dimensions.

Chapter 5 described a representation of single toleranced parts. A network was constructed that had nodes representing tolerance zones and datums. The arcs represent relationships and fall into four main categories. Firstly, the positions of the individual datums that make up a datum-system are constrained. Secondly, the zone of position tolerance of a feature is constrained relative to some specified datum-system. Thirdly, the zones that describe a single feature have their positions constrained by the fact that there must be room in their intersection for a real feature. Fourthly, the position of a datum is constrained relative to the zones of its associated feature. The last two types of relationship convert to inequality constraints whereas the first two convert to equalities. Representations for composite features were described. These allow tolerances to be applied and datums to be defined by regions of a part's surface larger than a simple feature. Composite features can be represented within the zone-datum network.

In chapter 6 assemblies of toleranced parts were dealt with. Again a network was constructed with nodes representing tolerance zones and datums. The networks arising from individual parts were linked by new arcs representing possible contacts between the parts. The geometry of contacts between toleranced features was formalised. The constraints contained two types of variable: those representing variation in shape and those representing variation in position.

7.2. Applications

The work has two main application areas: verification of tolerance specifications during design and uncertainty analysis in off-line robot programming.

Tolerance verification During Design

A tolerance specification must be satisfactory from the point of view of manufacture, economics and functionality. A lot of tedious work is involved in manually checking a tolerance specification. A computerised system could reduce this workload and allow more time for experimentation with different designs. The work as presented here can check tolerance specifications suggested by a designer.

Many design requirements can be expressed in terms of the sets of positions that could be obtained by the parts in the assembly. By deriving the set of positions attainable by each part in an assembly the work of this thesis allows such design requirements to be checked.

Suggestions will be made in the next section about how this work could actually be used to derive tolerance specifications that meet given requirements.

Robot Planning in the Presence of Uncertainty

Off-line progamming of robotic assembly provides the other application for this work. Automatic assembly planning is necessary in small batch robotic assembly so as to avoid wasting time with the robot out of action. Appreciation of the uncertainties involved allows sensing operations and other uncertainty reducing operations to be planned for. One of the most geometrically complex problems in analysis of the uncertainties occurs when many parts are in contact. Such is the case with a set of parts which have been partially assembled on the workbench. This problem can be handled by work in this thesis. The information obtained from this analysis can be integrated with knowledge of other sources of uncertainty to decide whether the next part can be inserted.

Other sources of geometric uncertainty in robotic assembly include (1) uncertainty in the position of the robot, (2) uncertainty in the position of parts provided to the robot and (3) uncertainty in sensor readings. The introductory chapter gave an example which gives insight into how robot position uncertainty could be handled. In a workable system different types of uncertainty should have compatible representations.

Another problem in robot planning is how to model uncertainty reduction. This occurs, firstly, when a sensor measures a quantity and secondly when new geometric relationships are created. In the former case the uncertainty in the measured quantity is reduced to the uncertainty of the sensor. An example of the latter occurs when a part is grasped. The uncertainty of the part is reduced by its newly formed relationships with the gripper.

7.3. Limitations and Possible Improvements

Some limitations of the work are discussed below along with improvements and extensions that could be made.

Small Uncertainties

Attention has been restricted to small uncertainties since these occur for many reasons such as the following.

- Manufacturing tolerances are always very small.
- Assemblies are rarely manufactured in which non-functional movement (slop) of a part is large.
- During a close fitting robotic assembly operation uncertainty must have been reduced somehow to a small level. However, since it is never possible to remove uncertainty altogether it is important to analyse the remaining uncertainty. It is reasonable to assume that uncertainty is small at this time.

The last point needs further discussion since, initially at least, uncertainties in the environment of a robot can be large. However, parts usually fit snugly and so precise movements are required to fit them together. Therefore, a robot plan always contains stages at which analysis of small uncertainties is appropriate.

Since, objects might be provided with a large amount of position uncertainty this must be reduced by, for example, a suitable sensor such as a camera or by creating new geometric relationships. After such operations the position of the part has a small uncertainty corresponding, for example, to the resolution of the camera or the accuracy of the new relationship. It is then appropriate to make an analysis which assumes uncertainties are small.

A planning system could contain packages for handling large uncertainty and small uncertainty. The former would insert steps into the plan that reduce uncertainty to a level where the latter can be applied.

Improved Representation of Uncertainty

A deficiency with the representation of uncertainty in this thesis is that it does not include statistical information. Quantities are given bounds but the distribution of the quantity within the bounds is not handled. Improved results could be obtained by attaching a statistical distribution to any range of values.

Suppose that a dimension has a 99% chance of lying within a range which is much smaller than that indicated by the bounds. It may be satisfactory if 1% of cases are ignored and so the effective range is the smaller one.

A case against using statistical distributions is that they are often difficult to determine in the first place. However, work by Bjørke summarised in chapter 2 shows how the manufacturing procedure can be used to predict distributions.

In many cases it is satisfactory to assume that square distributions are involved. However, the result of performing inferences on values with square distributions does not have a square distribution.

A representation of uncertainty could combine statistical distributions with upper and lower bounds. (It would be necessary to ensure that the integral of the distribution over the range is unity.) We would want to know how distributions combine. In this thesis two inferences - intersection and summation - were used for combining constraints. Perhaps analogous inferences could be found for uncertainties with statistical distributions though this would not be straightforward.

Approximate Results

Many results obtained by the analyses described in this thesis are approximate. It has been shown that certain types of approximate result are satisfactory from the point of view of determining functionality because they force a design to have less rather than more uncertainty. However, from the point of view of economic manufacture tolerances should not be much smaller than necessary. Therefore, approximations in results should be minimised.

Approximations arise from the following sources:

- Linearisation of rotations;
- Simplification of feature extents;
- Linearisation of constraints from curved features;
- Relationships have geometric constraints associated with them and algorithms have been described for converting these to algebraic constraints. However, in some cases the algorithms produce algebraic constraints which do not express the full effect of the geometric constraints.

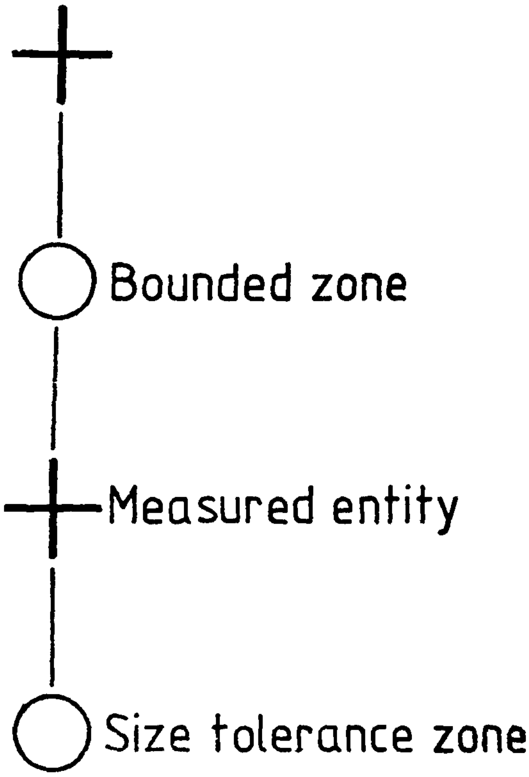
The first of these is small enough to be ignored if uncertainties are always small. The second and third are larger approximations but it can be ensured that they are over or underestimates as appropriate. They can only be removed at the expense of increasing the complexity of the constraints. It is difficult to know how significant these are without experimentation. The last type of approximation can only be removed by developing improved methods of obtaining the algebraic constraints from the geometry. It is not clear that such methods exist in general.

Other Types of Tolerance

Two types of tolerance defined by Requicha have been not been included in the computational representation of tolerances described in chapter 5. They are the RFS-position tolerance and curve tolerances formalised at the end of section 4.2. In Requicha's formalism a RFS-position tolerance is defined by a constraint on a measured entity (equivalent to a datum) defined by the feature. The measured entity lies inside a bounded zone with position constrained relative to some other datum. A curve tolerance is similar in many ways to a form tolerance but uses two-dimensional tolerance zones.

A suggestion for representing RFS-position tolerances is the following. Suppose a feature satisfies a size tolerance and a RFS-position tolerance (a feature satisfying only the latter would be underconstrained) then the network of tolerance zones and datums would be as shown in figure 7.3.1. The bounded zone is constrained relative to some datum. A measured entity of the feature is constrained relative to the bounded zone by the fact that it must lie inside it. The size tolerance zone is constrained relative to the measured entity. This is a similar type of relationship to that between a datum and a zone of the feature that defines the datum.

Curve tolerances do not seem to affect extremes of position or possible movements between parts and therefore would not affect the analyses described in this thesis. However, it might be possible to deal with curve tolerances by replacing them with combinations of other tolerances as was done with MMC-position tolerances (section 4.3).



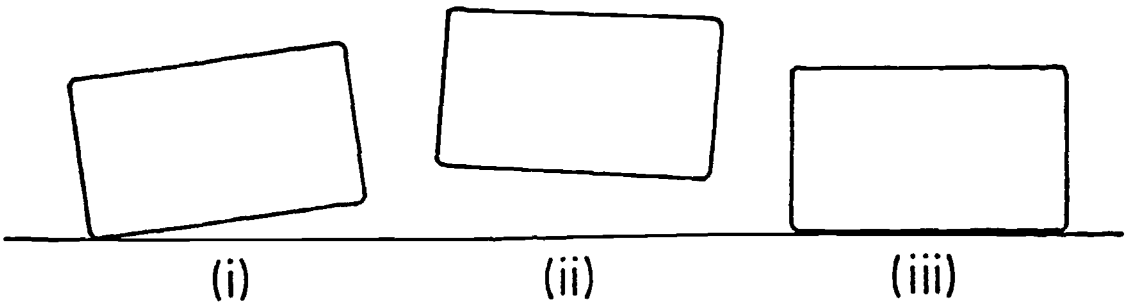
A suggestion for representing a RFS-position tolerance in the zone-datum network.

Figure 7.3.1

Forces and Fixtures

Throughout it has been assumed that the positions of parts are constrained only by the fact that the parts do not intersect. However positions are also affected by forces acting on the parts such as gravity and by fixtures such as bolts or glue. The result is that positions can be very much more constrained than is suggested by the condition that they do not intersect.

Suppose a block is resting on a table as shown in figure 7.3.2. The constraint that the block does not intersect the table does not prevent situations such as (i) and (ii). The presence of gravity, however, means that (iii) is the only plausible situation.

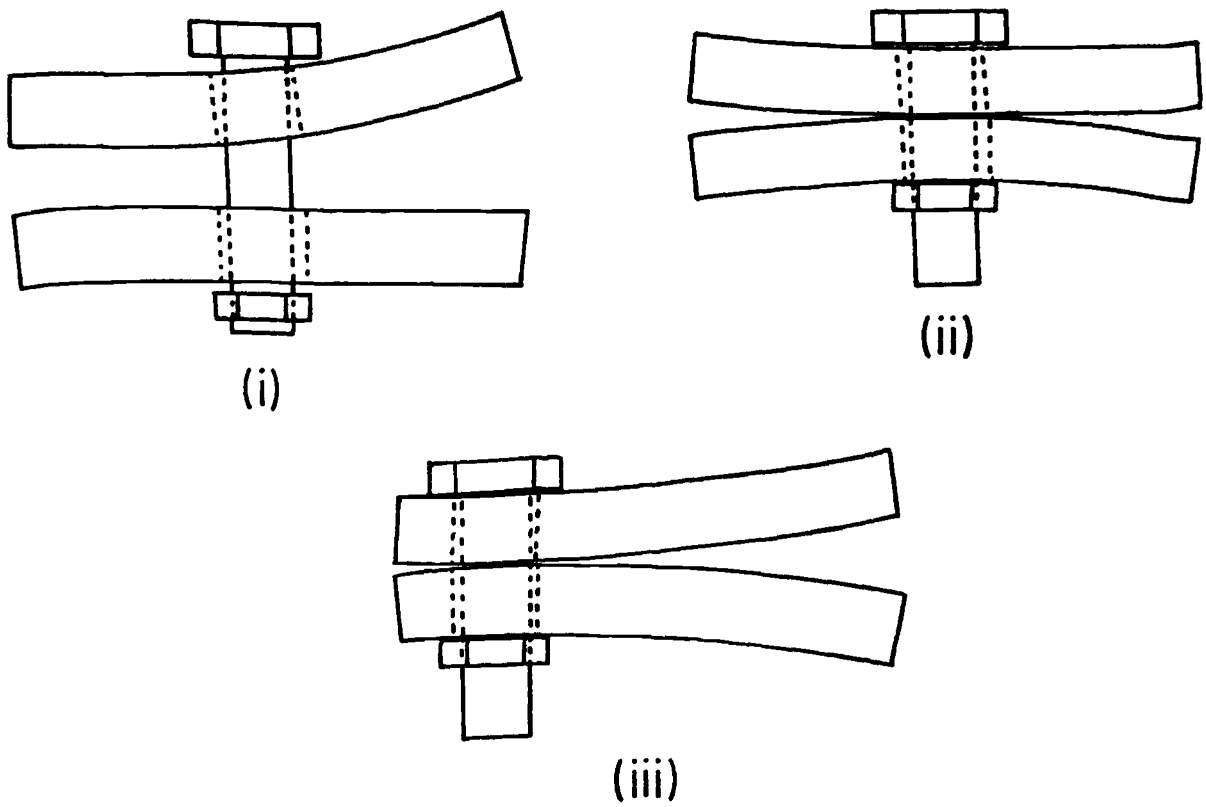


The position of the block is constrained by the block not intersecting the table.
(i) and (ii) Without gravity. (iii) With gravity.

Figure 7.3.2

Consider two nominally flat plates bolted together as shown in figure 7.3.3. As far as non-intersection is concerned their positions could be as in (i). However, if it is known that the bolt has been tightened then they must take on the positions shown in (ii). Figure 7.3.3(iii) demonstrates that the resulting position of the plates for these instances of the plates depends on the location of the bolt. Therefore, the way that their positions are constrained, over all possible instances, may also be dependent on the location of the bolt.

A possible method for handling these situations would be to change certain potential contacts into unconditional contacts taking note of the axis along which force is applied. In both examples, above, there is a force acting between the two parts. In the first case the force acts down from the centre of gravity of the block and in the second case it acts along the centre of the bolt.



Two nominally flat plates bolted together.
(i) Positions constrained simply by non-intersection of the parts.
(ii) Positions are further constrained by the fact that the tightened bolt asserts a force on the plates.
(iii) The resulting positions of these instances of the plates depend on the location of the bolt.

Figure 7.3.3

There is a problem with this method, however, in that, in general, it is difficult to decide which features are forced into contact. The actual surfaces that end up bearing the weight or force depend on the exact dimensions of the parts and so might vary from one part to another.

Another problem is to decide what positions are taken up by the plates when they are fixed by more than one bolt.

Glue causes an unconditional contact. The parts take up positions depending on the properties of the glue and how the parts were pressed together when the glue was applied.

In addition to creating an unconditional contact bolts and glue cause the position of the two parts to have fixed positions relative to one another. All relative degrees of freedom of the parts are removed.

Dynamic Structures

Only static structures have been considered in this thesis. As a result it has been possible to assume that a nominal position is known for each part. Dynamic situations such as the insertion of a part by a robot into a subassembly can be analysed as a series of static situations. However, there are effects that arise because the situation is truly dynamic. For example the amount of clearance needed between a peg being inserted into a hole might depend on the roughness of the surfaces of the peg and the hole. This effect has not been dealt with.

Derivation of Tolerance Specifications and Dimensions

All the work presented in this thesis has inferred unknown ranges of dimensions and positions from tolerances and ranges of positions which are known or can be calculated. This means that the variation in critical dimensions and positions can be evaluated and special requirements can be checked. However, we may want to make the reverse inference. Given requirements such as bounds on critical dimensions or on the relative position of certain parts we might want to know what tolerance allocation would guarantee these.

The former direction of inference allows a tolerance specification to be checked whereas the latter derives a tolerance specification from constraints imposed by the functionality of the object. In general the reverse inference is under-constrained and is therefore more difficult.

One method of handling the reverse inference would be to associate variables with dimensions or tolerance parameters whose values are unknown. Then the techniques described in this thesis could be applied to this parameterised geometry. The extra variables involved could be carried through the inferences without any difficulties except that the algebra would be more involved. Throughout this thesis it has been assumed that dimensions and tolerances have known values. Despite this, the theory could be applied to a parameterised geometry.

One practical difficulty with parameterising the geometry is that the algebra would be more complex since extra variables would appear in the constraints. However, interaction with an experienced user would help to keep the algebra at a manageable level. The user could select a few dimensions or tolerances to vary whilst keeping all others constant. In choosing appropriate dimensions or tolerances the user would make use of intuitive and unprogrammable knowledge. The system would be told which dimensions or tolerances to vary and the system would reply with constraints that must apply to these dimensions or tolerances if the design requirements are to be satisfied.

User interaction would also be useful for resolving problems in robot programming. Here, the user could make a wide variety of suggestions such as where in a plan to insert a sensing operation or an uncertainty reducing movement or possible design changes to the parts being manipulated. The system could reply with quantitative information about how accurate a particular sensor must be, what range of sizes a given dimension must have or what tolerance a particular feature must have.

7.4. Implementation

This section describes the requirements of a system based on the work in this thesis. Implementation of the work will initially take place as an enhancement to RAPT (section 3.1). The current extent of implementation, (section 3.7) covers the work described in chapter 3.

Features and Datums

The representation of parts should be compatible with the representation used by RAPT and should include all the information that RAPT needs to access. The basic components of their representation are surface features. The representation of features was described in section 3.1 but to make analysis of uncertainties as described in this thesis the addition of feature boundaries and tolerance attributes are required. Feature boundaries do not exist at present in models used by RAPT. However, RAPT has been linked to a CSG body modeller which contains more complete geometric models. The link has been used to create graphic output from RAPT. It is hoped that the same link will allow RAPT to extract boundary information from the modeller.

Each tolerance attribute includes (1) a name to indicate the tolerance type, (2) tolerance parameters and (3) in some cases a pointer to some datum system (section 5.2). The user could allocate these as part of the definition of each feature. Before a tolerance could be applied to a composite feature (section 5.10) the feature would have to be created by specifying a set of already created simple features. If editing were to be allowed then consistency checks would have to be performed. For example, it would make no sense to remove a simple feature if it were used in the definition of a composite feature.

Datums (section 4.2.3) are geometrically the same as RAPT features and so can be represented in exactly the same way. Each must contain a pointer to a feature which must be symmetric or planar. Each datum also has a nominal location in the part. To define a datum the user would input the datum type and the feature which defines it. In some cases this would be enough to define its nominal location. However, for some seco[illegible] datums it is also necessary to specify its relationship with the associated primary datum.

The Zone-Datum Network

The representation of a part consists of features and datums with tolerance attributes attached to features. Links exist between (1) a feature and the datum-system indicated by pointers in its position and orientation tolerance attributes and (2) a datum and its associated feature. By searching the representation for all situations where relationships between zones and datums occur a zone-datum network can be constructed (section 5.9.5). There must be a node in the network for each datum and a node for each tolerance attribute to represent a tolerance zone.

Assemblies of parts are described by listing the pairs of features between which a potential contact can occur. The zone-datum networks of individual parts are linked by arcs between each pair of zones associated with features potentially in contact.

Constraints must be attached to each arc of this network. Some of the information needed to derive the constraints can be catalogued because the possible number of different geometries is limited. This is particularly true for relationships whose constraints are equalities.

For example, the constraints between a zone of position tolerance and a datum (section 5.9.1) depend on the type of feature and datum and on their relative disposition: whether they are parallel or whether they intersect etc. The size of the catalogue of constraints for this type of relationship can be estimated as follows. There are three types of datum: planar, linear and point. The types of feature is small and four types, planar, cylindrical, conical and spherical might be sufficient. There are a few configurations of each type of zone relative to each type of datum which can be typically categorised as parallel, perpendicular, coaxial, coplanar etc. Hence, it is estimated that the number of cases in the catalogue of this type of relationship is of the order of 50.

Constraints for some types of relationships (sections 5.9.3, 5.9.4 and 6.1) can be calculated using signed distances (section 5.6). An algorithm must be available for calculating these (appendix 1). In the case of relationships in which just one feature is involved (for example, between a datum and zones of its associated feature) the algorithm must be given the extent of the feature. When two features are involved (as in the case of relationships representing potential contacts, section 6.1) the intersection of the extents of the features must be provided to the algorithm. The type of features involved is also important.

Analysis of the Constraints

Analysis of the network of constraints (sections 3.6, 5.11, 6.3 and 6.4) requires the ability to perform the following operations:

1. Find all paths between two chosen nodes in the network and reject those which are known to contribute no constraints.
2. Combine the constraints along each path.
3. Use the SUPINF algorithm to obtain results from the constraints (section 3.5 and appendix 2).

Paths between two nodes could be found using a depth first search. Paths should visit a node no more than once. As each link is added to the path currently being investigated the constraints could be combined with what has gone before. It might be found that after the constraints on a section of the path have been combined it is found that they are weaker than the constraints on a previous path. In this case the current path and all others which contain the same section can be ignored. In this way the number of paths to be considered can be reduced. A breadth first search would not allow this simplification to be made.

In conclusion, this section has discussed the issues involved with implementing the work in this thesis. A major portion of the work will be to create a suitable representation of geometry with convenient input and output. Use will be made of RAPT's representation of geometry along with the interface between RAPT and a body modeller. There would be consistency maintenance problems if editing were performed.

Implementation of the procedures for performing analyses described in this thesis should be straightforward. Catalogues of constraints would have to be constructed and an algorithm for finding signed distances would need to be created. Many of the constraint manipulation routines have already been implemented as described in chapter 3.

Appendix 1: SIGNED DISTANCE EXPRESSIONS

Signed distances were formalised in section 5.6. They were used for the evaluation of constraints in sections 5.9.3, 5.9.4, 5.10 and 6.1. In sections 5.9.3, 5.9.4 and 5.10 an expression is required which gives

$$\text{sdist}\left(O(H;d_1)\cap E \ , \ \text{comp}(O(H';d_2))\cap E \right) \qquad (1)$$

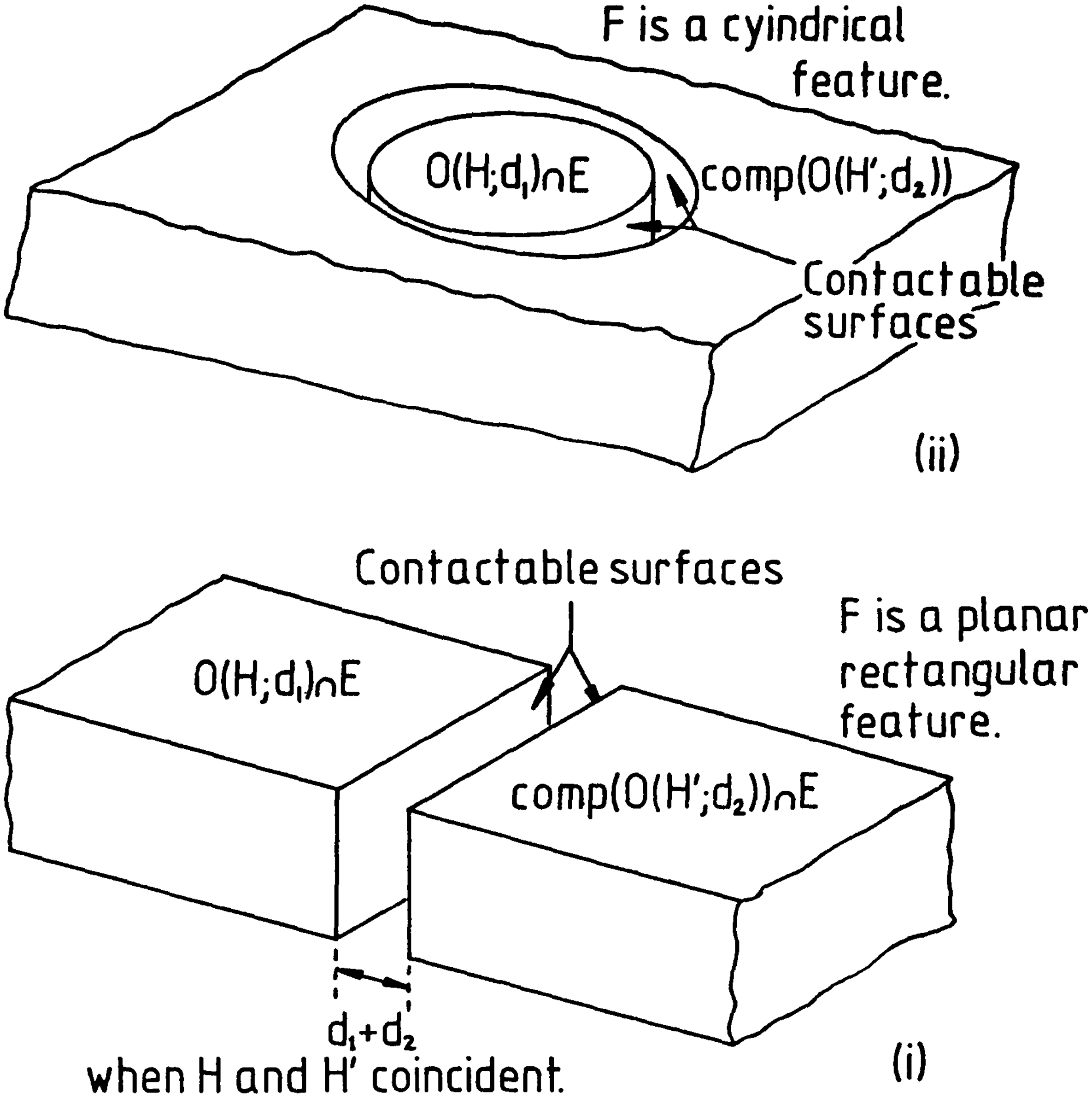
in terms of DOF-variables, where H is an extended feature of a nominal feature F, E is the extent-solid of F and H' is a congruent copy of H. d_1 and d_2 are numbers. The two solids involved in this expression are illustrated in figure A1 for a planar feature and a cylindrical feature. Under the small uncertainty assumption only the surfaces of these solids that correspond to the nominal feature can come into contact and so only these need to be considered.

In section 6.1 contacts between two different features are considered. Although it is assumed that the features have the same geometric type they may have different extents. However, matters can be simplified by projecting the extent of one feature onto the other and taking the intersection of the two extents. The result is called the extent of "overlap" of the features. It is possible to define an extent solid E for the overlap. Then the signed distance expression required in chapter 6 can be written

$$\text{sdist}\left(O(H_1;d_1)\cap E \ , \ \text{comp}(O(H_2';d_2))\cap E \right)$$

where H_1 and H_2 are the extended features of nominal features F_1 and F_2, H_2' is a congruent copy of H_2 and d_1 and d_2 are numbers.

An algorithm for finding the distance between two objects is given by Cameron (1984). This finds the distance between the



The solids involved in expression (1) when (i) F is a planar feature and (ii) F is a cylindrical feature.

Figure A1

closest points of the objects. Basically, it operates by finding the minimum distance between pairs of items, one item from each object, with the following types:

face-face	face-edge	face-vertex
edge-edge	edge-vertex	vertex-vertex.

The minimum of all the distances thus obtained is the distance between the objects.

In practice, the task is made easier by finding the minimum distances between unbounded versions of the faces and edges. The minimum distance between two items occurs between two points on the items. These are called the "closest points". If the closest points of the unbounded versions of two items are found to be outside the boundary of the items then the closest distance between the **bounded** versions of the items will be found when considering the distances between the boundary elements of the items. This follows from the fact that the closest point on a face to some other item either occurs on the face itself or on one of the edges that bounds the face. The closest point of an edge to another item either occurs on the edge itself or on a vertex that bounds the edge.

This algorithm can be applied to the evaluation of signed distances. Signed distances can be evaluated by looking at the same pairs of types of item.

In the problem of interest here, several simplifications can be made to the algorithm for the following reasons: (1) we know which pair of faces of the objects come close to one another so all other faces can be ignored, (2) the faces have the same geometric shape, (3) the two faces are bounded in the same way and (4) the objects only deviate a small distance from their nominal position.

The problem is reduced to finding the signed distance between the pair of faces indicated as being able to contact in figure A1. Each face has a material side and an air side. The signed distance between a pair of faces is, by definition, negative if their material sides intersect, zero if the faces touch and positive otherwise. The signed distance between an edge (or a vertex) and a face is similarly negative if the edge (or vertex) intersects the material side of the face and zero or positive otherwise.

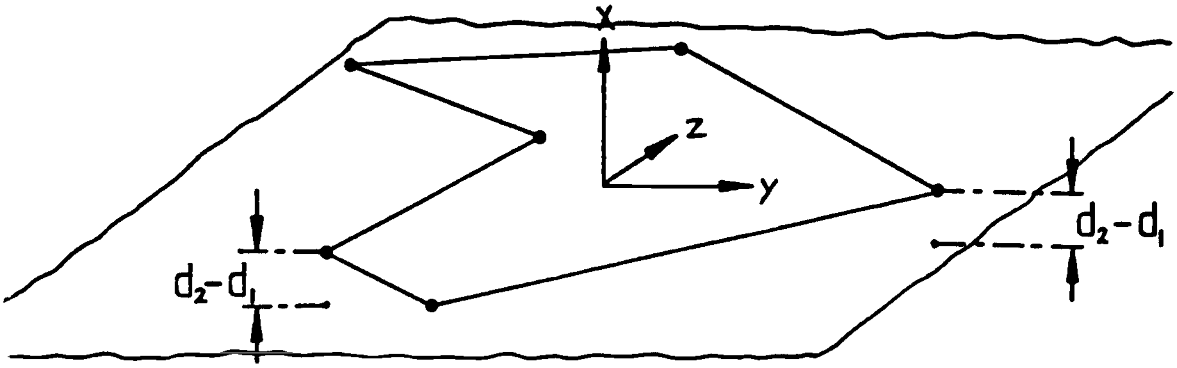
Denote the faces between which the signed distance is to be evaluated by A and B. One of the faces, say A, can be considered to be unbounded without making any difference to the signed distance between A and B. This is because the faces only deviate a small distance from their nominal positions: the region over which they overlap is approximately the same as the extent of the faces themselves. As a result the edges and vertices that surround A can be ignored. Hence instead of evaluating signed distances between all of the pairs of item types given above it is sufficient to evaluate signed distances between the following pairs of items.

- Unbounded face A and unbounded face B.
- Unbounded face A and unbounded edges of B.
- Unbounded face A and vertices of B.

The minimum should be taken of signed distances between these pairs of items rejecting certain pairs as follows.

- The pair consisting of face A and face B should be rejected if the closest point on B to A falls outside the boundary of face B.
- A pair consisting of face A and an edge of B should be rejected if the closest point on this edge of B to A falls outside the boundary of the edge of B.

Examples will be given of how signed distances can be evaluated between solids arising from planar features and cylindrical features. This will be followed by a description of an algorithm that could evaluate a signed distance expression for a feature with any given geometric type and extent.



Surfaces arising from a planar feature with polygonal extent. The unbounded version of A and the bounded version of B are shown. Nominally the planes are separated by d_2-d_1.

Figure A2

Example - Planar features

In the case of planar features with polygonal extent (figure A2) it turns out that we only need to consider signed distances between one face A (considered unbounded) and the vertices of the other face B. Signed distances between face A and face B and between A and edges of B fall into the categories for rejection given above.

Suppose that when A and B are at their nominal positions then they are both parallel with the yz-plane. From expression (1) it is seen that the nominal separation of A and B is d_2-d_1. Suppose that, at their nominal positions, A passes through the origin and B is coincident with the plane $x=d_2-d_1$. Suppose that there are n vertices of B with coordinates (d_2-d_1,y_i,z_i) when B is at its nominal position $(i=1,\ldots,n)$. The material side of A has $x<0$ and the material side of B has $x>d_2-d_1$.

When A and B are at their nominal positions the signed distance between A and each vertex of B is d_2-d_1. If B is moved by an amount given by DOF-variables $(\delta x,\delta y,\delta z,\delta\theta,\delta\phi,\delta\psi)$ then vertex i is displaced by

$$(\delta x, \delta y, \delta z)+(\delta\theta, \delta\phi, \delta\psi)\times(0, y_i, z_i),$$

where × represents the vector product. We are interested in the component of this vector along the normal of A. It is

$$\delta x + z_i\delta\phi - y_i\delta\psi.$$

Therefore, the signed distance between A and vertex i of B is

$$d_2 - d_1 + \delta x + z_i\delta\phi - y_i\delta\psi.$$

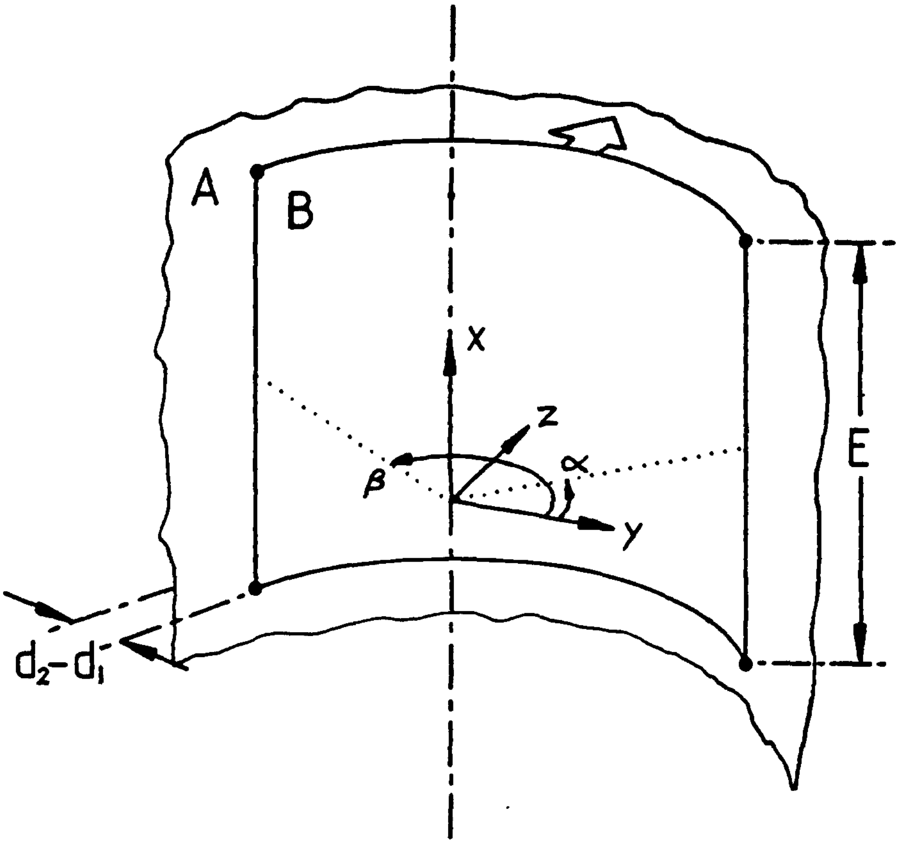
The signed distance between A and B is the minimum of these expressions for i=1,...,n. Thus,

$$\text{sdist}\big(\, O(H;d_1)nE \;,\; \text{comp}(O(H';d_2))nE \,\big) = d_2 - d_1 + \delta x + \min_{i=1,..,n}\big(z_i\delta\phi - y_i\delta\psi\big)$$

Example - Cylindrical Features

Suppose that the feature F is a segment of a cylinder (figure A3). The extent of the cylinder is defined by a length along its axis and by the angle that it extends around its axis. A and B are cylindrical surfaces with their axes nominally coincident with the x-axis. Suppose that the length of the cylinder is E and that it is bounded by planes $x=\pm E/2$. Suppose that the cylindrical feature extends from an angle α to an angle β about the x-axis measured from the y-axis.

The faces A and B are each bounded by four edges, two of which are straight lines and are parallel with the x-axis and two of which are circular arcs centred on the x-axis. There are four vertices where these edges meet.



Cylindrical surfaces arising from a cylindrical feature. The unbounded version of A and the bounded version of B are shown. The angular extent of the feature is defined by the angles α and β measured from the y-axis. The upper circular arc on the boundary of B undergoes a translation in the yz-plane of ($\delta y+E\delta\psi/2$, $\delta z+E\delta\phi/2$).

Figure A3

The signed distance between A and B can be found by considering signed distances between,

- the unbounded version of A and the unbounded versions of the circular edges of B,
- the unbounded version of A and the vertices of B.

Signed distances between faces A and B and between A and the straight edges of B can be rejected.

Let us first consider the signed distance between the unbounded circular arcs of B and unbounded face A. The x-direction can be ignored during this process. Vectors written with two components will represent directions in the yz-plane. Angles in this plane will be measured relative to the positive y-axis which is given angle zero.

One circular arc has x=E/2 and the other has x=-E/2. Nominally the signed distance between one of the arcs and surface A is d_2-d_1. Suppose that B is moved from its nominal position by $(\delta x,\delta y,\delta z,\delta\theta,\delta\phi,\delta\psi)$. The arc with x=E/2 undergoes the displacement (figure A3),

$$(\delta y,\delta z)+(E/2)(\delta\psi,\delta\phi) = (\delta y+E\delta\psi/2,\delta z+E\delta\phi/2).$$

The point on the arc which ends up being closest to A depends on the direction of this vector:

$$\tan^{-1}\big((\delta y+E\delta\psi/2)/(\delta z+E\delta\phi/2)\big).$$

If this is between α and β then the closest point on the arc to A lies between the bounds of the arc. In this case the signed distance between the bounded arc and the unbounded version of A is given by

$$d_2-d_1-\sqrt{(\delta y+E\delta\psi/2)^2+(\delta z+E\delta\phi/2)^2}$$

If the direction is not between α and β then we must determine the signed distance between A and the vertices at the ends of this arc. The vertices are displaced by the same vector as the arc on which they lie:

$$(\delta y+E\delta\psi/2,\delta z+E\delta\phi/2).$$

However, we must find the component of this vector in the direction of the normal of the surface of A taken near to the relevant vertex. For the vertex with angle α the normal is $(\cos\alpha, \sin\alpha)$ and so the required component is

$$(\delta y+E\delta\psi/2, \delta z+E\delta\phi/2).(\cos\alpha, \sin\alpha) = \cos\alpha(\delta y+E\delta\psi/2)+\sin\alpha(\delta z+E\delta\phi/2)$$

where "." is the scalar vector product. Therefore the signed distance of the vertex from A is

$$d_2-d_1-\cos\alpha(\delta y+E\delta\psi/2)-\sin\alpha(\delta z+E\delta\phi/2).$$

Similarly the signed distance to A from the vertex with angle β is

$$d_2-d_1-\cos\beta(\delta y+E\delta\psi/2)-\sin\beta(\delta z+E\delta\phi/2).$$

Similar expressions are obtained for the edge and vertices with $x=-E/2$ by replacing E with -E.

Finally, the resulting signed distance between A and B can be expressed as min(P,Q) where P and Q are given by the following:

If $\alpha \leq \tan^{-1}\left((\delta y+E\delta\psi/2)/(\delta z+E\delta\phi/2)\right) \leq \beta$

then let $P = d_2-d_1-\sqrt{(\delta y+E\delta\psi/2)^2+(\delta z+E\delta\phi/2)^2}$

otherwise let $P = \min_{\gamma=\alpha,\beta}\left[\; d_2-d_1-\cos\gamma(\delta y+E\delta\psi/2)-\sin\gamma(\delta z+E\delta\phi/2) \;\right].$

If $\alpha \leq \tan^{-1}\left((\delta y-E\delta\psi/2)/(\delta z-E\delta\phi/2)\right) \leq \beta$

then let $Q = d_2-d_1-\sqrt{(\delta y-E\delta\psi/2)^2+(\delta z-E\delta\phi/2)^2}$

otherwise let $Q = \min_{\gamma=\alpha,\beta}\left[\; d_2-d_1-\cos\gamma(\delta y-E\delta\psi/2)-\sin\gamma(\delta z-E\delta\phi/2) \;\right].$

In General ...

An algorithm to calculate a signed distance for any shape of feature could be implemented using these techniques. It would have to know how to calculate signed distances between the following pairs of items:

1. Two unbounded faces;

2. An unbounded edge and an unbounded face;

3. A vertex and an unbounded face.

It would also have to be able to determine the closest points of these pairs of items and determine if they are within the bounds of the items.

Catalogues can be produced of signed distances between the pairs of items given above. This would require classification of the types of faces and edges that are involved. Faces have the same set of types that simple nominal features have (eg. planar, cylindrical, spherical and cylindrical). For each type of face there would be a set of possible edge types that could bound that face. In the planar feature example above only one type of edge was involved: all edges were linear. In the cylindrical feature example there were two types of edge: linear, parallel to the cylinder's axis and, circular, about the cylinder's axis.

Hence it follows that signed distances between the pairs of items given above only need to be known for a small number of cases.

The algorithm would also have to have to work out the closest point of an unbounded face or unbounded edge to another unbounded face. It would have to check that this point lies within the bounds of the face or edge. The required information could be catalogued according to type of face and type of edge. The catalogue would contain procedures for calculating:

1. Whether the closest point of two faces lies within the bounds of the face where the face is bounded by edges restricted to a few types.

2. Whether the closest point of an edge to a face is within the bounds of that edge.

Appendix 2: Algorithms SUP and INF

A complete specification of the algorithms for SUP and INF is given by Brooks in (1981). Here, a summary of these algorithms is given which shows how they can be used to handle linear expressions and constraints.

Basically these algorithms find the bounds on a variable or an expression subject to a set of inequality constraints. The algorithms take three arguments. The first is an expression whose bounds are to be determined. The second argument, C, is a set of inequality constraints that must be satisfied by the variables occurring in the expression of the first argument. The third, V, is a set of variables which may occur in the bounds of the expression output by the algorithm. If this set is empty then numeric bounds will be determined. Examples of the result of calling SUP with a variety of different third arguments are given in section 3.5.

First define UPPER(A,C) and LOWER(A,C) where A is a variable and C is a set of inequality constraints. UPPER(A,C) is an expression which does not contain A and which bounds A above. LOWER(A,C) is similar but bounds A below.

UPPER(A,C) is obtained directly from the inequalities in C as follows. For each inequality in C containing A solve that inequality for A. Collect together all expressions obtained in this way which bound A above (rejecting expressions which bound A below) and denote them by E_1, E_2, ..., E_n. Then UPPER(A,C) = $\min(E_1,E_2,\ldots,E_n)$.

Also define LOWER(A,C) as an expression which bounds A below and which is obtained from the inequalities in C in a similar way.

The following table shows the output from SUP if its three arguments are denoted by A, C and V. The right hand column contains actions that are performed if A satisfies the corresponding condition in the left hand column. The definition of INF can be obtained by making the textual substitutions "SUP" → "INF", "INF" → "SUP", "UPPER" → "LOWER", "min" → "max", "≤" → "≥" and "∞" → "-∞".

Condition	Actions
1. A is a number	Output A
2. A is a variable contained in V.	Output A
3. A is a variable not contained in V.	Let B=SUP(UPPER(A,C), C, V U {A}). a. If A does not occur in B then output B. b. If A does not occur in B then solve A≤B for A to get A≤B′ where A does not occur in A. Output B′.
4. $A = A_1+A_2$	Output $SUP(A_1,C,V)+SUP(A_2,C,V)$.
5. $A = A_1-A_2$	Output $SUP(A_1,C,V)-INF(A_2,C,V)$.
6. $A = min(A_1,A_2)$	Output min(SUP(A,C,V) , SUP(B,C,V)).
7. A = rB where r is a number.	If r>0, output rSUP(B,C,V). If r<0, output rINF(B,C,V).
8. A matches none of the above.	Output ∞.

In effect, variables occurring in the expression in the first argument of SUP but which do not occur in its third argument are replaced by expressions which bound the variables above. However, the expression used to replace a variable may itself contain variables that need replacing and so SUP must be applied recursively to this expression. To prevent an infinite loop the replacement expression must not include variables that have already been replaced. This process is represented by step 3 above. If A consists of an expression involving summations, differences or minimums of terms then SUP is applied to each of the terms in the expression. This is represented by steps 4, 5 and 6.

Example

Suppose C is the set of inequalities

$$\{ 1+y \le x \le 10-y \;,\; 5-y \le x \le 6+y \}.$$

The evaluation of SUP(x-y,C,{x}) would proceed as follows.

SUP(x-y,C,{}) =
SUP(x,C,{}) - INF(y,C,{})

UPPER(x,C) = min(10-y,6+y).
SUP[min(10-y,6+y),C,{x}] =
min(10-INF(y,C,{x}) , 6+SUP(y,C,{x})) =

LOWER(y,C)=max(x-6,5-x).
INF(max(x-6,5-x),C,{x,y}) = max(x-6,5-x).
So, INF(y,C,{x}) = max(x-6,5-x).

UPPER(y,C)=min(x-1,10-x).
INF(min(x-1,10-x),C,{x,y}) = min(x-1,10-x).
So, SUP(y,C,{x}) = min(x-1,10-x).

min(10-max(x-6,5-x) , 6+min(x-1,10-x)) =
min(16-x,5+x,5+x,16-x) =
min(16-x,5+x).
Solving x ≤ min(16-x,5+x) gives x≤8.

So, SUP(x,C,{}) = 8.

LOWER(y,C) = max(x-6,5-x).
INF(max(x-6,5-x),C,{y}) =
max(INF(x,C,{y})-6 , 5-SUP(x,C,{y})) =

LOWER(x,C) = max(1+y,5-y).
INF(max(1+y,5-y),C,{y,x}) = max(1+y,5-y).
So, INF(x,C,{y}) = max(1+y,5-y).

UPPER(x,C) = min(10-y,6+y).
SUP(min(10-y,6+y),C,{y,x}) = min(10-y,6+y).
So, SUP(x,C,{y}) = min(10-y,6+y).

max(max(1+y,5-y)-6 , 5-min(10-y,6+y)) =
max(y-5,-1-y,y-5,-1-y) =
max(y-5,-1-y).
Solving y ≤ max(y-5,-1-y) gives y ≤ -0·5.
So INF(y,C,{}) = -0·5.

Therefore, SUP(x-y,C,{}) = SUP(x,C,{}) - INF(y,C,{})
= 8-(-0·5) = 8·5.

References

(Ambler et al 1983)
"Reasoning about the Spatial Relationships Derived from a RAPT Program for Describing Assembly by Robot" by A.P. Ambler, D.F. Corner and R.J. Popplestone, IJCAI-83, Karlsruhe, West Germany, 1983.

(Bjørke 1976)
"Computer-Aided Tolerancing" by Ø. Bjørke, Tapir, Trondheim, Norway.

(Brooks 1981)
"Symbolic Reasoning Among 3-D Models and 2-D Images" by R.A. Brooks, Memo AIM-343, Dept. of Computer Science, Stanford University, 1981.

(Brooks 1983)
"Symbolic Error Analysis and Robot Planning" by R.A. Brooks, Memo AIM-685, Dept. of Computer Science, Stanford University, 1983.

(BS 308)
British Standards, BS:308, Part 2 - "Dimensioning and tolerancing of size" and Part 3 - "Geometric tolerancing"

(Cameron 1984)
"Modelling Solids in Motion" by S.C. Cameron, Ph.D. Thesis, Dept. of Artificial Intelligence, University of Edinburgh, 1984.

(Clocksin et al 1982)
"Progress in Visual Feedback for Robot Arc-Welding of Thin Sheet Metal" by W.F. Clocksin, P.G. Davey, C.G. Morgan and A.R. Vidler, pp189-200, 2nd Int. Conf. on Robot Vision and Sensory Control, Stuttgart, Germany, 1982.

(Dufay and Latombe 1984)
"An approach to Automatic Robot Programming Based on Inductive Learning" by B. Dufay, and J.C. Latombe, The International Journal of Robotics Research, Vol 3 No 4, 1984.

(Fleming 1985a)
"Analysis of Uncertainties in a Structure of Parts" by A. Fleming, AISB-85, Warwick, U.K., pp54-61, 1985.

(Fleming 1985b)
"Analysis of Uncertainties in a Structure of Parts" by A. Fleming, IJCAI-85, Los Angeles, pp1113-1115, 1985.

(Hillyard 1978)
"Dimensions and Tolerances in Shape Design" by R. Hillyard Tech. Report No. 8, Computer Laboratory, University of

Cambridge, 1978.

(Hoffman 1982)
"Analysis of Tolerances and Process inaccuracies in Discrete Part Manufacturing" by P. Hoffman, Computer-Aided Design, Vol. 14 No. 2, 1983.

(Michael and Siddall 1981)
"The Optimal Tolerance Assignment with Less than Full Acceptance" by W. Michael and J.N. Siddall, Journal of Mechanical Design, Vol. 104, pp855-860, 1982.

(Lozano-Pérez et al 1984)
"Automatic Synthesis of Fine-Motion Strategies for Robots" by T. Lozano-Pérez, M.T. Mason and R.H. Taylor, The International Journal of Robotics Research, Vol 4 No. 1, 1984.

(Lozano-Pérez and Wesley 1979)
"An algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles" by T. Lozano-Pérez and M.A. Wesley, Communications of the ACM, Vol 22 No. 10, 1979.

(Parkinson 1984)
"Tolerancing of Component Dimensions in CAD" by D.B. Parkinson, Computer-Aided Design, Vol. 16 No. 1, 1984.

(Requicha 1977)
"Mathematical Models of Rigid Solid Objects" by A.A.G. Requicha, Tech. Memo 28, Production Automation Project, University of Rochester, 1977.

(Requicha 1983a)
"Toward a Theory of Geometrical Tolerancing" by A.A.G. Requicha Tech. Memo No. 40, Production Automation Project, University of Rochester, 1983.

(Requicha 1983b)
"Representation of Tolerances in Solid Modelling: Issues and Alternative Approaches" by A.A.G. Requicha, Tech. Memo 41, Production Automation Project, University of Rochester, 1983.

(Requicha and Chan 1985)
"Representation of Geometric Features, Tolerances and Attributes in Solid Modellers Based on CSG" by A.A.G. Requicha and S.C. Chan, Tech. Memo 48, Production Automation Project, University of Rochester, 1985.

(Popplestone and Ambler 1981)
"A Language for Specifying Robot Manipulations" by R.J. Popplestone and A.P Ambler, DAI Research Paper No. 161, Dept. of Artificial intelligence, University of Edinburgh, 1981.

(Taylor 1976)
"A Synthesis of Manipulator Control Programs from Task-Level Specifications" by R.H.Taylor, Memo AIM-282, Stanford Artificial Intelligence Laboratory, Stanford University, 1976.

(Yin 1984)
"Combining Vision Verification With a High-Level Language" by B. Yin, Ph.D. Thesis, Dept. of Artificial Intelligence, University of Edinburgh, 1984.