15

Modelling of Vague and Precise Geometric Information for Supporting the Entire Design Process

X. Guan and K. J. MacCallum
University of Strathclyde
CAD Centre, Department of Design, Manufacture and Engineering Management, University of Strathclyde, 75 Montrose Street, Glasgow G1 1XJ, UK. Telephone: +44-141-5524400. Fax: +44-141-5523148.
email: x_guan@cad.strath.ac.uk, ken@cad.strath.ac.uk

Abstract

One of the key problems in building a computer aided design system that supports the entire design process is the development of a suitable computational scheme that enables the representation and manipulation of product geometric information with an evolving degree of vagueness or precision from early to detail stages of the design process. This paper reports the initial results arising from our ongoing research into the solution to this problem. A scheme that integrates techniques of constraint handling, interval representation and geometric modelling is developed. It is used to facilitate the representation and manipulation of such information in geometric configuration design. It is shown that geometric constraints provide a 'natural' language for describing various types of geometric information used in design particularly in the early stages. This scheme provides a uniform representation for the types of vague and precise information used in our pilot study and unifies their processing. Further research is being undertaken to enhance its abilities in accommodating more complex situations and supporting evolving design towards its completion.

Keywords

Early design support system, computer aided geometric design, geometric configuration, computer aided design

T. Tomiyama et al. (eds.), *Knowledge Intensive CAD*

1 INTRODUCTION

In spite of the advances since the early 1960s, Computer Aided Design (CAD) systems still require a lot of research effort in a number of areas such as providing assistance to the entire design process, in particular the early stages of this process, improving the capability of modelling various types of product information, and better integration with computer aided manufacturing systems (Nielsen, Dixon and Simmons, 1987, Shah and Wilson, 1989, Tovey, 1989). The research reported in this paper mainly contributes to the first of these areas. It addresses the issues involved in the handling of geometric information concerning a product from its early conception to the detail development.

While precise and concrete information is available and must be used to define the geometric properties of a product completely and uniquely at the detail design stages, at the early stages where precision is not of great concern, there exists a mixture of geometric information that is either vaguely or precisely known. Thus, one of the key problems to resolve in building a CAD system that assists the entire product design process is the development of a suitable computational scheme that enables the representation and manipulation of such geometric information whose degree of vagueness or precision evolves during the process of design. In other words, a scheme that enables a minimum commitment modelling principle (Guan, 1993).

This paper presents the initial results arising from our ongoing research into such a scheme. Section 2 presents a characterisation of the geometric information dealt with in design with a focus on the early stages. Thinking underlying our scheme is laid down in Section 3. A classification of the various types of geometric constraints is established in Section 4. Section 5 describes the representation structure and reasoning mechanism established for modelling product geometric configuration. Implementation of these ideas in building a prototype support system is also discussed. Finally, a discussion of the scheme is given in Section 6.

2 CHARACTERISATION OF PRODUCT GEOMETRIC INFORMATION

Geometric information is the set of facts that are specified and used to describe or derive the geometric properties of a product required for its manufacturing. Since geometric information available at the detail design stages (characterised by its precision, concreteness and completeness) has been covered and modelled well by existing CAD systems, in this section we focus on that used at the early stages of the design process.

Geometric information at early stages of design can be classified into four types: shape, size, location and orientation.

- *Shape* The shape of an object may be described as 1D, 2D, or 3D generic primitives. For example, as the characteristics for searching design variants, Pahl and Beitz (1988) provided a list of such primitive shapes: `curve, circle, ellipse, ..., triangle, square, rectangle, ..., cylinder, cone, rhomb, cube, sphere`. It may also be complex such as those shown in Figure 1 which is a collection of shape variants explored in a final year student project in the authors' department - redesign of a steam/spray iron model (Gaddis, 1987). Since design may be carried out through feature development, the shape of an object may also be defined through specific design features. Use of these features, such as `rib`, `wall`, `hole`, `boss`, `projection`, `depression`, `gusset`, can also be observed in design guidelines and rules (Guan, 1993). All of these features imply particular types of shape.

- *Size* The size, or dimension, of an object may be given in the form of relations among various design parameters such as `width`, `depth`, etc. These relations are usually various types of linear (sometimes non-linear) inequalities, equalities as well as ranges. For example, in the design specification of a motor vehicle fuel gauge given in (Pahl and Beitz, 1988) (page 60), the `volume` and `height` of the container are given to be from $20l$ to $160l$ and from $150mm$ to $600mm$, respectively. Many guidelines and rules practised in design also specify relations or constraints on the various design parameters. For instance, in the case of plastic injection-moulding design, the following relational patterns can be observed from the various guideline and rules (Guan, 1993):

$$x \geq \mathrm{a}y, \tag{1}$$
$$x \leq \mathrm{a}y, \tag{2}$$
$$x \geq \mathrm{a} + \mathrm{b}y, \tag{3}$$
$$x = \mathrm{a}y + \mathrm{b}z, \tag{4}$$
$$x = [\mathrm{a}y, \mathrm{b}z], \tag{5}$$
$$x \leq \min\{\mathrm{a}y, \mathrm{b}z\}. \tag{6}$$

where x, y, z denote design parameters such as `thickness`, `radius`, `diameter`, `width`, `depth`, `height`, `angle`, and `distance`. The most basic patterns of relations are $x\Delta\mathrm{a}$ and $x = [\mathrm{a}_1, \mathrm{a}_2]$, where $\Delta \in \{\leq, <, \geq, >, \approx, =\}$, a, a_1, and a_2 are real numbers, and [,] denotes a value range.

- *Location and Orientation* The location, or position, and the orientation of an object are related to the spatial arrangement of a product. They may be described explicitly in spatial relationships such as `connect to`, `attach to`, `below`, `parallel`, `perpendicular` as observed in the guidelines and rules practised in design (Guan, 1993). They are, however, more often expressed implicitly in design sketches in relation to other objects or chosen datum, as can be found in (Tjalve, 1979, Hubka, Andreasen and Eder, 1988). Figure 2 shows one such example from (Hubka, Andreasen and Eder, 1988).

Clearly, early geometric information can be imprecise which presents a range of possible choices, such as sizes specified in terms of some of the constraints described earlier, or locations specified in terms of spatial relationships. It may also be an abstraction or simplification of a larger set of information and thus represents this larger set by implication. As a partial model of a geometric object, such a simplification usually represents some structural, basic, or interesting aspects of the object's geometry, e.g. convex-hull, bounding-box, centre-line of a hole, or the mid-point of the centre-line (Woodbury and Oppenhelm, 1987). Thus, geometric information at the early stages of design may be a mixture of vague and precise information caused by poor or vague design specification, a lack of information or knowledge, and a designer's desire for minimum commitment to unnecessary precision and details in order not to reduce the design solution space.

Figure 1 A collection of iron shape concepts explored in (Gaddis, 1987).

3 AN APPROACH TO THE HANDLING OF GEOMETRIC INFORMATION

It is clear that development of a suitable computational scheme that is able to represent and manipulate geometric information with an evolving degree of vagueness or precision is a key problem encountered in building a CAD system that supports the entire design process. Based on the observation that relations of various types are frequently used in design to describe the size, location and orientation of objects which are not uniquely or precisely specified, we envisage that it may be possible to establish a scheme for handling the types of geometric information characterised above by integrating the method of constraint management, certain numeric approximation handling technique and the techniques of geometric modelling. A subsequent assessment of existing uncertainty handling methods revealed that the interval algebra based method (Moore, 1979) could serve the

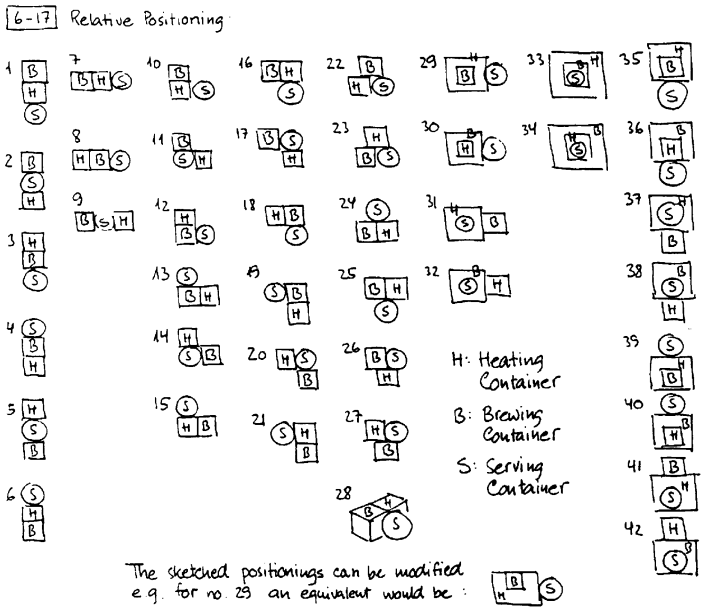


Figure 2 Location and orientation information in some of the basic spatial arrangements from (Hubka, Andreasen and Eder, 1988).

purpose of our initial investigation by providing a direct, straightforward and simple way of representing both approximate and precise numeric values (Guan, 1993).

Based on the above vision, our approach to the modelling of the above geometric information is to characterise the geometry of an object by geometric parameters separated into: *size parameters* which describe the size of an object, such as `width`, `depth` and `height` of a cuboid object, *location parameters* that are used to characterise the position of the object in an arrangement, such as the `centroid` of the object, and *orientation parameters* that further describe the arrangement attribute of the object. Approximate and precise geometry can thus be described by parameters that have approximately or precisely defined values. The values of these geometric parameters are derived, using the techniques of constraint reasoning, from high-level geometric constraints given by designers, such as those discussed in the previous section, and are represented uniformly by real interval numbers.

In the rest of the paper, we first present a classification of such high-level geometric constraints (Section 4) and then illustrate the use of the above approach in building a geometric configuration design support system (Section 5). Finally, a discussion of the approach will be given (Section 6).

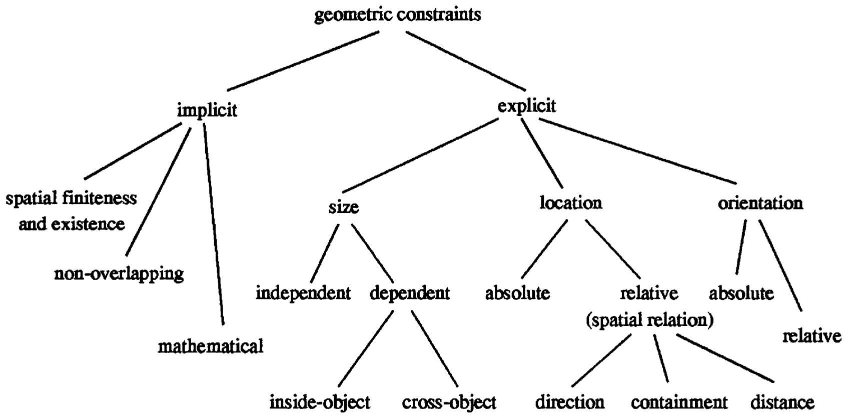


Figure 3 A classification of geometric constraints.

4 CLASSIFICATION OF GEOMETRIC CONSTRAINTS

Figure 3 shows a structure for classifying high-level geometric constraints that may be used in a system as a language for modelling product geometry.

4.1 Implicit Geometric Constraints

Implicit constraints reflect the human being's understanding and knowledge of the geometry of the world. They are divided into three types:

- *Spatial Finiteness and Existence Constraint* An artefact, in particular solids considered in engineering design, is finite in space and occupies a certain volume.
- *Non-Overlapping Constraint* Two individual, i.e. distinctive, solid objects should not interfere with each other spatially.
- *Mathematical Constraint* This consists of relations or formulae that capture the inherent properties of a certain geometric shape and can thus be used to compute the geometric properties of an object of that shape.

4.2 Explicit Geometric Constraints

An explicit geometric constraint is a constraint specified directly by a designer or derived from the designer's specification. It may have attached to it an importance factor and a time stamp. An *importance factor* is a number between, say, 0 and 1 inclusively, where 0 indicates the least important constraint and 1 the most important. A *time stamp* records the time when the associated constraint is established in the design process. Both the importance factor and the time stamp are useful in resolving inconsistent or conflicting constraints.

Explicit geometric constraints are further classified into size, location and orientation constraints.

Size Constraints

A size constraint specifies a relation that needs to be satisfied by one or more size parameters. It is defined here as an arithmetic relation of the form $t_l \Delta t_r$, where t_l and t_r are arithmetic terms in which size parameters are variables, and Δ is an arithmetic relation symbol defined here as one of $\{=, \approx, \neq, >, \geq, <, \leq\}$. An *arithmatic term* is a variable, a numeric constant or a compound term constructed using arithmetic function symbols {+, -, *, /, sin, cos, max, min, abs, pow}, variables and numeric constants. For example, $\texttt{radius}_1 \approx 100$, $\texttt{radius}_2 = [58, 62]$, $\texttt{depth} + \texttt{width} \leq 150$, $\texttt{height}=\max\{\texttt{depth},\texttt{width}\}$ are all examples of size constraints admitted by the above definition.

Size constraints are further divided into two types: dependent and independent constraints.

- *Dependent Size Constraints* A dependent size constraint is one that contains at least two different size parameters. For instance, the following are three such constraints:

$$\texttt{depth}_\texttt{A} + \texttt{height}_\texttt{A} \geq \texttt{width}_\texttt{A}, \tag{7}$$
$$\texttt{depth}_\texttt{A} = \texttt{depth}_\texttt{B}, \tag{8}$$
$$\texttt{height}_\texttt{A} \approx \texttt{height}_\texttt{B}/1.8. \tag{9}$$

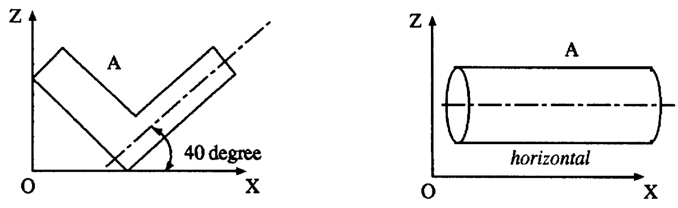
  where $(\texttt{width}_\texttt{A}, \texttt{depth}_\texttt{A}, \texttt{height}_\texttt{A})$ and $(\texttt{width}_\texttt{B}, \texttt{depth}_\texttt{B}, \texttt{height}_\texttt{B})$ are the size parameters of two objects A and B of cuboid shape, respectively.
  A dependent size constraint can be an inside-object or a cross-object constraint. An *inside-object constraint* is one in which all the involving size parameters describe the same object, such as (7) of the above. A *cross-object size constraint*, on the other hand, is one in which the size parameters belong to different objects, such as (8) and (9) of the above.
- *Independent Size Constraints* An independent size constraint is a size constraint with only one size parameter. The following is a set of possible independent size constraints:

$$\texttt{width}_\texttt{A} = 80, \tag{10}$$
$$\texttt{height}_\texttt{B} \leq 78, \tag{11}$$
$$\texttt{depth}_\texttt{B} \approx 57, \tag{12}$$
$$\texttt{width}_\texttt{B} = [75, 78]. \tag{13}$$

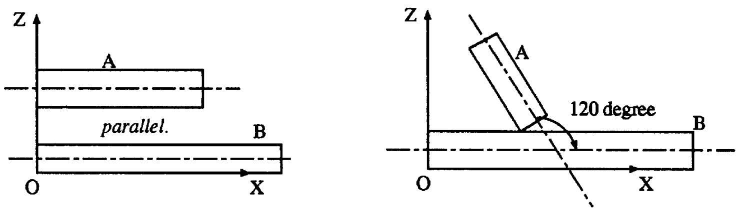
Location Constraints

Location constraints specify the possible locations of an object. Two types are distinguished here: absolute and relative. An *absolute location constraint* defines the position of an object using a coordinate system OXYZ as reference. It specifies some or all of the x, y, z coordinates of a position at which the chosen location parameter of an object lies. A *relative location constraint*, which is also called a spatial relation, defines the position of an object in relation to other objects. Currently, the following types of spatial relations are identified:

- *Direction Constraints* Constraints of this type specify the location of an object in relation to another one along a defined direction. Examples are `above`, `below`, `right`, `left`, `behind`, `front`, etc.



(a) Absolute constraints



(b) Relative constraints

Figure 4 Examples of orientation constraints.

- *Distance Constraints* Constraints of this type specify the distance of an object in relation to other objects.
- *Containment Constraints* Constraints of this type can be used to specify the location of an object in relation to another one so that spatially it contains, or is contained by, the other object.

Orientation Constraints

Orientation constraints are those which specify the orientation of an object in a spatial arrangement. They may be absolute which gives orientation in relation to a coordinate system through some datum or may be given relatively in terms of other objects through chosen datum, as illustrated in Figure 4 (3D right-handed Cartesian coordinate systems are used).

5 MODELLING OF PRODUCT GEOMETRIC CONFIGURATION

To investigate the feasibility and capability of the approach outlined earlier for accommodating product geometric information with varying degree of vagueness, we have developed a scheme for supporting the design of product geometric configuration based on the approach. Here, *geometric configuration* refers to the process of exploring the approximate or precise geometry of individual components of a product and the spatial arrangements of these components in forming the total product geometric structure (Guan, Stevenson and MacCallum, 1995). We have further limited the pilot study to the class of geometric configuration problems based on only simple geometric information: 3D primitive shape {`cuboid, sphere, cylinder, frustum, prism`}, independent size constraints, six basic directional spatial relations {`above, below, right, left, behind, front`} as well as fixed point position. This section describes the scheme in terms of the representation structure and reasoning mechanism.

5.1 Representation Structure

Major elements of the conceptual structure for representing the geometric configuration is presented in Figure 5 using the NIAM (Nijssen's Information Analysis Methodology) notations (Verheijen and van Bekkum, 1982). In this structure, a *geometric configuration world* consists of all geometric configuration alternatives that are being investigated (simultaneously) for a product. A *geometric configuration* consists of one or many *geometry* entities that are spatially related to one another. Each of the geometry entities represents a geometric model of a component of a product being designed. Each geometric configuration is constrained spatially by a *geometric configuration space* that is defined as a 3D cubic space with a given 3D right-handed Cartesian world coordinate system OXYZ. Geometric configuration space reflects one of the implicit geometric constraints discussed earlier (*Spatial Finiteness*), and is used here as a physical bound for geometric configuration design. Similar to the value of a geometric parameter to be discussed later, a geometric configuration space is represented by three ranges along the three axes of the coordinate system. Here, 'range' - from a *range* of value - is used as a synonym of interval. An interval is a bounded set of real numbers represented by a lower and an upper bound (Moore, 1979). Each *geometry* entity consists of inherent attributes of *shape* and *size*, and arrangement attributes of *location* and *orientation* of a component being represented. The shape of a component may be any of the primitives {`cuboid, prism, cylinder, frustum, sphere`}. The size of a component is characterised by size parameters. The value of a size parameter can be defined approximately or precisely which is represented by a range (interval). It is given by a designer through a set of *size constraints*.

The location of a component in a geometric configuration is characterised by a location parameter, called *datum point* which is a point in the component. This datum point lies in a 3D cubic uncertain region which captures the approximation or uncertainty associated with the location of the component. An uncertain region is represented by three ranges (intervals) along, respectively, the X, Y, and Z axes of the world coordinate system associated with the corresponding geometric configuration which specify the allowable x, y, and z coordinates of the corresponding datum point. The bounds of the uncertain region are determined by location constraints such as spatial relations {`above, below, right, left, behind, front`} (Figure 6) with other components or a fully specified point position in the world coordinate system. Boundary models can be constructed to represent a component. A boundary model refers to the geometric model constructed via the boundary representation scheme developed in the field of geometric modelling.

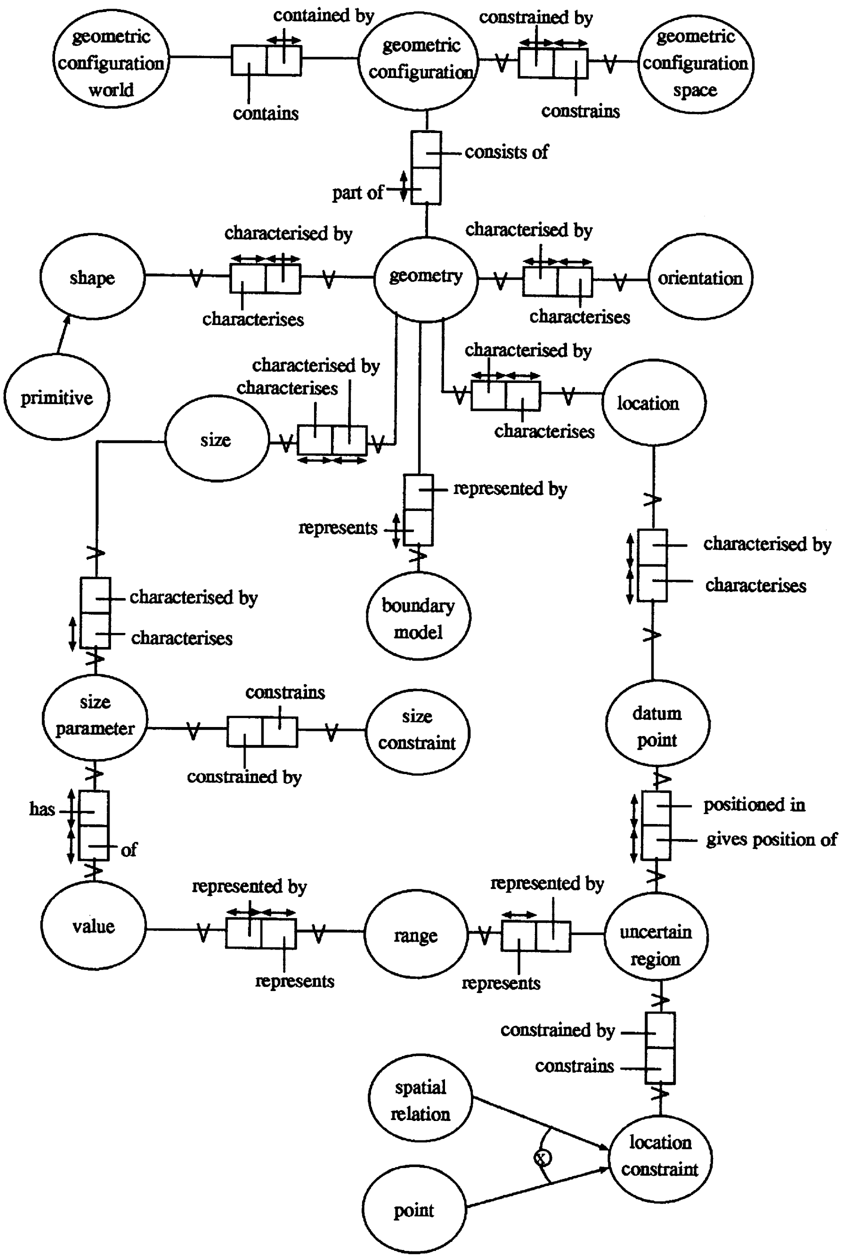


Figure 5 Major elements of the overall representation structure of geometric configuration.

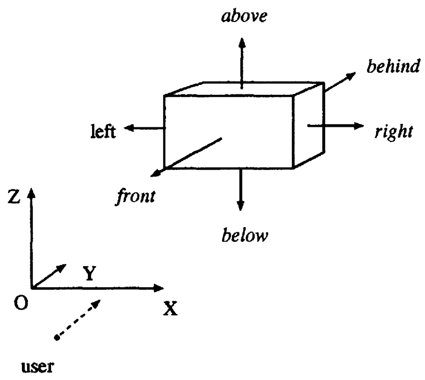


Figure 6 A set of basic spatial relations.

5.2 Reasoning Mechanism

To support the manipulation of geometric configuration models represented through the above structure, we have developed a reasoning mechanism which can be viewed as consisting of three layers as illustrated in Figure 7. This mechanism treats the problem of constructing and manipulating geometric configurations as mainly that of establishing and satisfying relevant geometric constraints, combined with interval handling and geometric modelling. Each layer is composed of directed processes and a layer in a higher level leads to that below. The clouds in the figure denote the major type of information/data manipulated in each layer.

The Constraint Layer

The first layer of this mechanism is constraint management consisting of the tasks of establishment of constraints, inconsistency detection and resolution, and solving of constraints explained below.

Establishment of Constraints This process involves the construction of constraint models on the size and location parameters of components as part of the geometric configuration model being established in the system from a designer's specification. For size parameters, constraints of inequality and/or equality types are directly established by a supporting system based on the designer's specification. The location of a component in a configuration is given by a designer in spatial relations or a point position and is represented by a datum point which lies in an uncertain region. To find the bounds of the uncertain region, low-level constraints on these bounds are formulated from the spatial relations or point position.

Handling of Inconsistency It is possible that a new size or location specification for an object in the form of size and location constraints conflicts or is inconsistent with the existing geometric model of the component or with other components in the same configuration. It is therefore necessary to detect the possible conflicts and inconsistencies, and to resolve them.

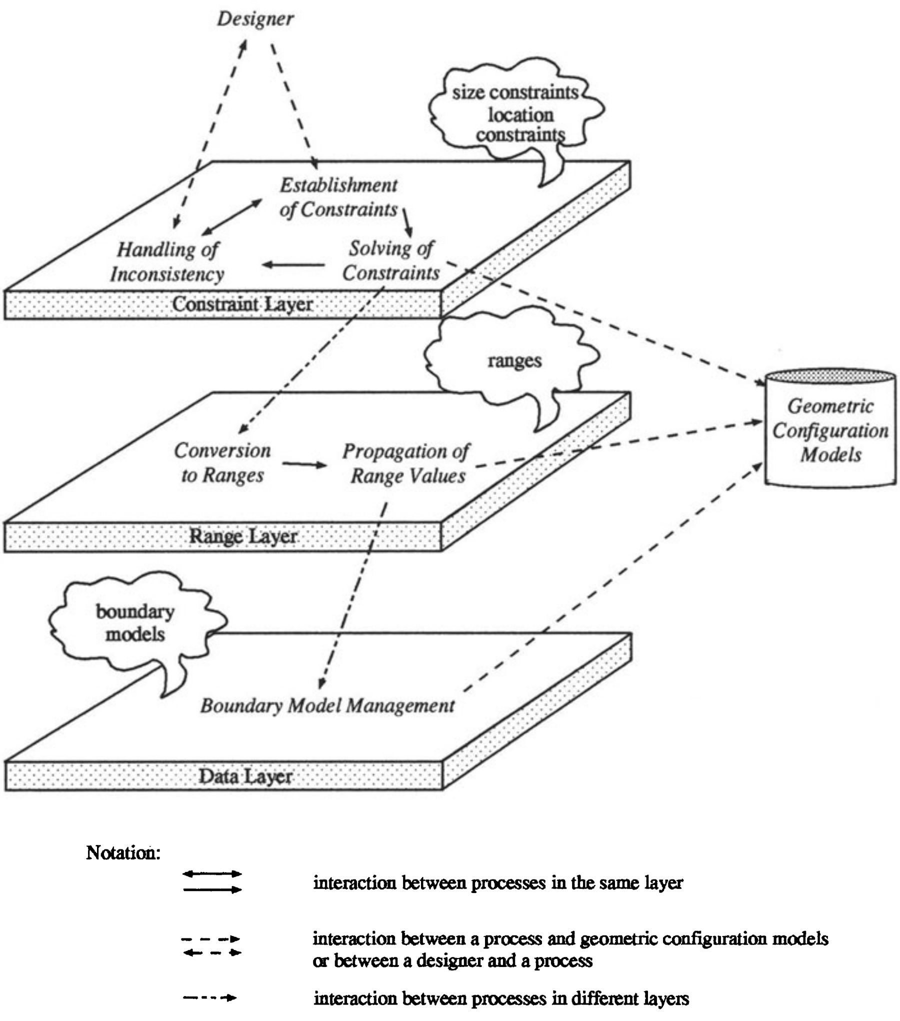


Figure 7 A reasoning mechanism.

Solving of Constraints This task is required for solving the size constraints specified by the designer and the low-level constraints on the bounds of the uncertain regions formulated from location constraints such as spatial relations or point positions. This involves finding appropriate values for size or location parameters defined by the set of constraints. If no solutions can be found for these constraints, the resolution process will be invoked.

Table 1 Range conversion

Constraints	Value
$x = \mathrm{a}$	$[\mathrm{a}, \mathrm{a}]$
$x < \mathrm{a}$	$[\mathrm{new_{min}}, \mathrm{a})$ or $[\mathrm{default_{min}}, \mathrm{a})$
$x \leq \mathrm{a}$	$[\mathrm{new_{min}}, \mathrm{a}]$ or $[\mathrm{default_{min}}, \mathrm{a}]$
$x > \mathrm{a}$	$(\mathrm{a}, \mathrm{new_{max}}]$ or $(\mathrm{a}, \mathrm{default_{max}}]$
$x \geq \mathrm{a}$	$[\mathrm{a}, \mathrm{new_{max}}]$ or $[\mathrm{a}, \mathrm{default_{max}}]$
$x \approx \mathrm{a}$	$[\mathrm{a_{min}}, \mathrm{a_{max}}]$, $\mathrm{a_{min}} = \mathrm{a} - \mathrm{degree_{approx}}/2$ $\mathrm{a_{max}} = \mathrm{a} + \mathrm{degree_{approx}}/2$
$x = [\mathrm{a_1}, \mathrm{a_2}]$	$[\mathrm{a_1}, \mathrm{a_2}]$

The Range Layer

This layer converts solution derived in the Constraint Layer (the solving process) into ranges that represent the approximate or precise values of the corresponding geometric parameters, and propagates the resulting range values to the whole geometric configuration model.

Conversion to Ranges This process transforms the results from the solving process of the Constraint Layer to ranges. A range is an interval described by a lower and an upper bound. Let x denote a geometric parameter, and a, $\mathrm{a_1}$ and $\mathrm{a_2}$ denote real numbers. Table 1 lists the conversion used. Since it can be expressed by others, the $\neq$ type is not included in our initial study. The conversion of $x = \mathrm{a}$ and $x = [\mathrm{a_1}, \mathrm{a_2}]$ is straightforward as shown in the table. Since $x\Delta\mathrm{a}$ where $\Delta \in \{<, \leq, >, \geq\}$ only defines one value bound for x, a default minimum and maximum value, denoted by $\mathrm{default_{min}}$ and $\mathrm{default_{max}}$ respectively ($0 < \mathrm{default_{min}}, \mathrm{default_{max}} <$ the corresponding dimension of the geometric configuration space, as imposed by the implicit geometric constraint - *Spatial Finiteness and Existence*) are introduced here to provide the missing bound in case that the designer does not want to or cannot supply a value. $\mathrm{default_{min}}$ and $\mathrm{default_{max}}$ are provisional which can be changed by the designer as required. In the case where a minimum value or a maximum value is supplied, it is denoted by $\mathrm{new_{min}}$ or $\mathrm{new_{max}}$, respectively. The round brackets ')' and '(' in the ranges for $x < \mathrm{a}$ and $x > \mathrm{a}$ respectively mean that the number a is not included in the corresponding value ranges*. To convert $x \approx \mathrm{a}$, a default degree of approximation, $\mathrm{degree_{approx}}$, which is greater than or equal to zero is introduced. This $\mathrm{degree_{approx}}$ is used as the width of the interval as shown in the table.

Propagation of Range Values This process propagates the converted value ranges to the related geometric configuration model. Two levels of propagation may be distinguished: propagation across different components which are related to one another through certain size, location or orientation constraints (*propagation in breadth*), and propagation among the representation entities of the same components (*propagation in depth*).

*Practically, exclusion of a in $x < \mathrm{a}$ or $x > \mathrm{a}$ may be treated as that the upper bound of $x < \mathrm{a}$ or the lower bound of $x > \mathrm{a}$ takes $\mathrm{a} - \delta$ or $\mathrm{a} + \delta$ respectively, where δ is a very small positive number in relation to a.

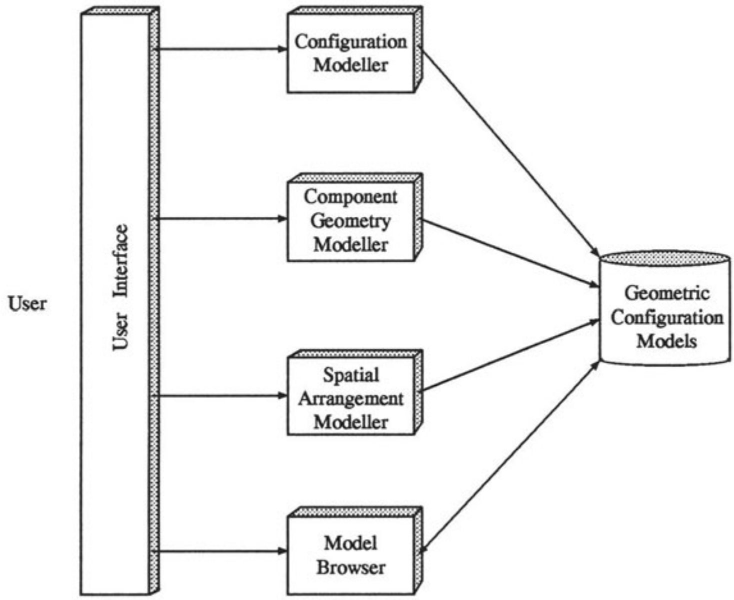


Figure 8 Overall structure of the system.

Data Layer

Boundary models are created when necessary to provide concrete geometric model for the individual components of a configuration as described in the representation structure. This layer, therefore, provides the necessary schemes for management of these models. It consists of the processes of constructing and updating the boundary models using the values resulting from the Range Layer.

Since a lot of available geometric modelling systems provide boundary representation, this layer can be established by building the necessary interface to such a system. The geometric modelling system can thus be called through the interface whenever construction or modification is made in the layer above (the Range Layer).

5.3 Implementation

The above representation structure and reasoning mechanism have been embedded in a prototype system. From the user's point of view, this system can be regarded as consisting of the components shown in Figure 8. Through the User Interface, a designer constructs, modifies and inspects graphically or textually various Geometric Configuration Models using the utilities provided by the Configuration Modeller, the Component Geometry Modeller, the Spatial Arrangement Modeller, and the Model Browser. Main features of the current system are:

- Use of the primitives - {`cuboid, cylinder, sphere, prism, frustum`} - to represent the shape of an object.

- Specification of the size of an object via size constraints of inequalities, e.g. `height` $\leq$ 13.2, `depth` $\approx$ 12.7, ranges e.g. `width` = [20.95, 22.12], or equalities e.g. `width` = 15.49.
- Establishment of spatial arrangement of the objects via the six spatial relations - {`above, below, right, left, behind, front`} - or fixed point positions.
- Reduction of size and location approximation by incremental refinement. For example, if it was given earlier that `width` = [20.95, 22.12], and now new information `width` $\leq$ 21.50 is added to the same component, then the value range of width will be reduced to [20.95, 21.50].
- Constraint reasoning and solving by the system, and inconsistency resolution through the cooperation of the user and the system. Currently the system can handle simple inconsistency or conflict such as when `A above B` and `A below B` are given to A at the same geometric configuration, or when `width` = [20.95, 21.50] and `width` $\leq$ 19.95 are given to the same object.
- Exploration of alternative geometric configurations which may have different spatial relationships and approximate or precise sizes.

The system has been implemented on a SunSparc platform using CLOS (Common Lisp Object System), a generic constraint solver, CLP(R) (Heintze, et al, 1991), and a geometric modeller, ACIS (Spatial Technology Inc, 1992). Figure 9 shows a snapshot of the system during a geometric configuration session. For more descriptions of the system, see (Guan, Stevenson and MacCallum, 1995).

6 DISCUSSION

Constraint handling has been used in variational/parametric geometry systems (Light and Gossard, 1982, Solano and Brunet, 1994). Compared with traditional geometric modelling based systems, these variational geometry systems provide better support to designers. They are descriptive since they permit the user to construct design drawings and to specify certain geometric, mostly dimensional, constraints on them without worrying about the sequence of operations (Parden and Newell, 1984). Dimensioning of drawings and the subsequent modification of dimensions provide a more natural and convenient way of specifying and working on the geometry of objects in design.

Essentially, in variational geometry systems, the geometry of an object is represented by a set of characteristic points. Dimensional and other constraints, such as tangency, are treated as the defining relations between the chosen characteristic entities on the object, and are interpreted as non-linear and linear equations on the coordinates of the characteristic points. The extensive computational effort involved in solving a set of such equations has limited the application of the technique to complex 3D objects. In the approach presented in this paper, the geometry of an object is characterised by meaningful geometric design parameters. High-level constraints specified by the designer are solved to obtain the values of these parameters. These values are not used further to establish non-linear or linear equations on characteristic points as in the variational geometry systems. Instead, they are kept explicitly and used directly, when necessary, as an input to a geometric modelling system to construct the boundary models. Thus, constraint handling is used in our approach to provide a language for describing the geometry of an object and the interface between the designer and the underlying geometric modelling system.

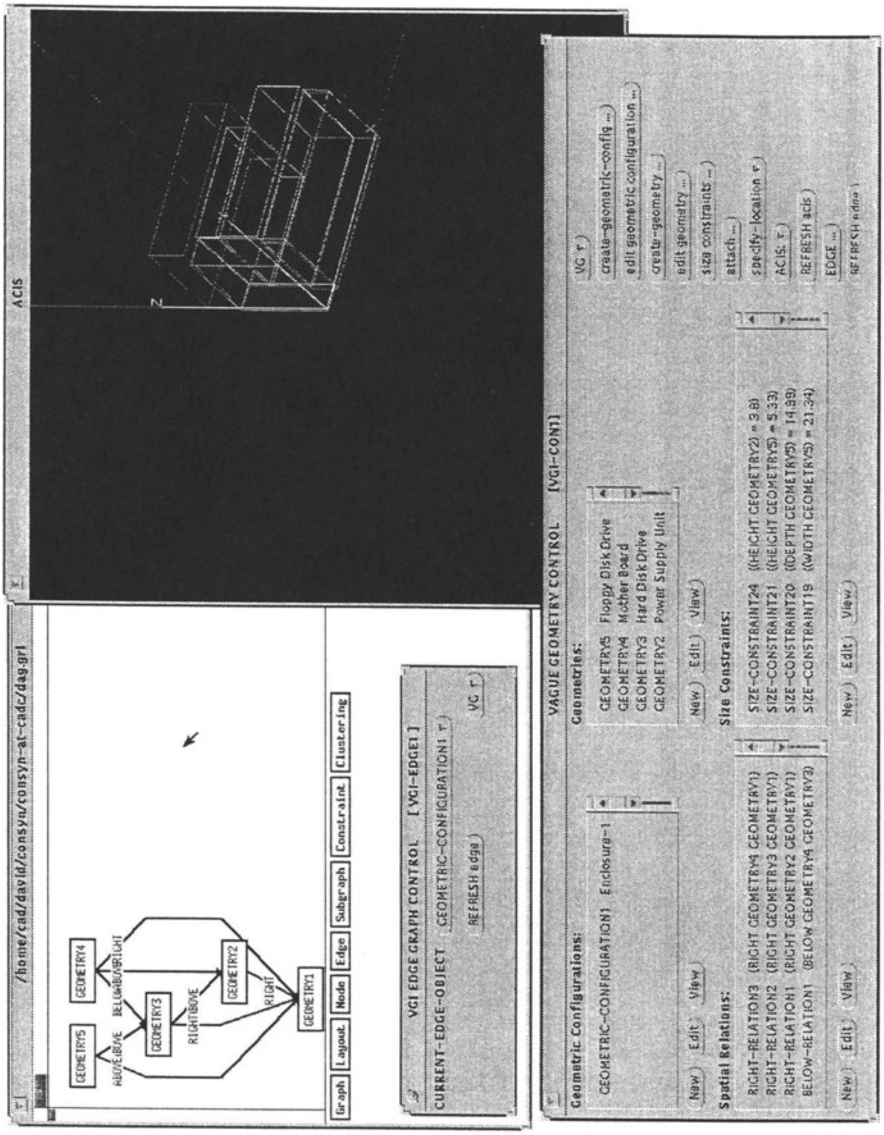


Figure 9 A snapshot of the system in a geometric configuration session.

Pahl, G. and Beitz, W. (1988) *Engineering design: a systematic approach.* The Design Council, London.

Parden, G. and Newell, R.C. (1984) A Dimension Based Parametric Design System, in *CAD'84*, Sussex, U. K.

Shah, J.J. and Wilson, P.R. (1989) Analysis of design abstraction, representation and inferencing requirements for computer-aided design. *Design Studies*, **10**, 169-78.

Solano, L. and Brunet, P. (1994) Constructive constraint-based model for parametric CAD systems. *Computer-Aided Design*, **26**, 614-21.

Spatial Technology Inc. (1992) *ACIS: interface guide.*

Tjalve, E. (1979) *A short course in industrial design.* Newnes-Butterworths.

Tovey, M. (1989) Drawing and CAD in industrial design. *Design Studies*, **10**, 24-39.

Verheijen, G.M.A. and van Bekkum, J. (1982) NIAM: An Information Analysis Method, in *Information Systems Design Methodologies: A Comparative Review* (eds. T.W. Olle, H.G. Sol and A.A. Verrijn-Stuart), North-Holland.

Woodbury, R.F. and Oppenhelm, I.J. (1989) An Approach to Geometric Reasoning, in *Intelligent CAD, I* (eds. H. Yoshikawa and D. Gossard), North-Holland.

10 BIOGRAPHY

Xiaohong Guan is a research assistant at the CAD Centre, University of Strathclyde, UK. Her current research interests include early design support systems, geometric reasoning, constraint management and geometric modelling. She obtained a BSc in Radio Electronics Science from University of Sichuan, China, in 1984, an MSc in Information Engineering from Xidian University, China, in 1987 and a PhD from University of Strathclyde, UK in 1994. She is a member of AAAI, IEEE and IEEE Computer Society.

Professor Ken MacCallum obtained his first degree in Naval Architecture from the University of Glasgow, proceeding to postgraduate study in Imperial College, University of London where he obtained a PhD for research into the application of computer graphics to free-form surface design. After three years with a software company, he joined the University of Strathclyde, establishing the CAD Centre in 1985 as a research and postgraduate centre. He is currently the Head of Design, Manufacture and Engineering Management in the Faculty of Engineering at the University of Strathclyde. Ken MacCallum's main area of research has been the application of Artificial Intelligence to Engineering Design. He has led projects concerned with intelligent design modelling, data exchange, computer based design coordination, and computer aided learning. He is editor of the International Journal on Artificial Intelligence in Engineering, is a member of IFIP WG5.2, and has been on the Technical Programme Committees of a large number of Conferences and Workshops concerned with computer aided design.

spatial configurations. Consequently, improvements on the interval representation, or alternative representations, require to be investigated to cope with uncertain regions with complex shapes.

Other interesting issues resulting from our research include the detection and resolution of geometric inconsistencies, incorporation of design features, and integration with other product information such as tolerances, cost, etc.

7 SUMMARY

A scheme for modelling geometric information with an evolving degree of vagueness has been presented in this paper in response to the requirements of supporting the entire design process. This scheme integrates the techniques of constraint handling, interval representation and geometric modelling. It uses high-level geometric constraints as an 'interfacing language' between the designer and a support system to model geometric design information. It provides a uniform representation for the types of vague and precise information used in a pilot study, and further unifies their processing. Further research issues have also been identified.

8 ACKNOWLEDGEMENT

This work is supported by EPSRC, U. K.

9 REFERENCES

Gaddis, F. (1987) *Design proposal: Rowenta DA72.1 - redesign.* Final-Year Student Project Report, Design Division, Department of Design, Manufacture and Engineering Management, University of Strathclyde, U. K.

Guan, X. (1993) *Computational support for early geometric design.* PhD Thesis, Department of Design, Manufacture and Engineering Management, University of Strathclyde, U. K.

Guan, X., Stevenson, D.A. and MacCallum, K.J. (1995) A Prototype System for Early Geometric Configuration, in *Proceedings of the 3rd International Conference on Computer Integrated Manufacturing* (eds. J. Winsor, A.I. Sivakumar and R. Gay), World Scientific, Singapore.

Heintze, N., Jaffar, J., Michaylov, S., Stuckey, P. and Yap, R. (1991) *The CLP(R) programmer's manual,* version 1.1. IBM Thomas J. Watson Research Centre, USA.

Hubka, V., Andreasen, M.M. and Eder, W.E. (1988) *Practical studies in systematic design.* Butterworths.

Light, R. and Gossard, D. (1982) Modification of geometric models through variational geometry. *Computer-Aided Design,* **14**, 209-14.

Moore, R.E. (1979) *Methods and applications of interval analysis.* SIAM, Philadelphia.

Nielsen, E.H., Dixon, J.R. and Simmons, M.K. (1987) *How shall we represent the geometry of designed objects?* Technical Report 6-87, Department of Mechanical Engineering, University of Massachusetts.

Although they are very useful in investigating families of parts that have the same shape or topology but different size and for making subsequent changes on size, variational geometry systems seem not to provide suitable means for exploring spatial arrangements of multi-component products qualitatively. Rapid and qualitative spatial arrangement is a very significant element of early geometric design where a designer investigates the structural or topological organisation of the product without committing to unnecessary details. To support this task, we have incorporated location constraints in our approach. The pilot study has indicated that high-level location constraints (even the six most basic spatial relations) make a very positive contribution towards this aspect.

Although they support a level of abstraction higher than conventional geometric modelling based systems, variational geometry systems do not provide a means of recording and using approximate information. Use of the interval method in our research has provided a uniform underlying representation of both approximate and precise numeric values. The pilot study has demonstrated its feasibility in representing the approximation introduced by inequality type of independent size constraints and the six spatial relations. Furthermore, use of intervals facilitates a minimum commitment interpretation of these constraints without unnecessarily reducing the solution space perceived by the designer.

To investigate the feasibility and capacity of the approach proposed in this paper in handling more complex geometric configurations, research in the following aspects is being carried out:

- Supporting the transition and evolving of approximate and abstract geometric models to precise and detailed ones by supporting different levels of abstraction.
- Handling the other types of geometric constraints shown in Figure 3, such as dependent size constraints and other spatial relations.

Studies are also required to establish the suitability of the interval representation method used in our approach and to identify alternative methods. It is our belief that the interval representation is quite a natural interpretation and description of the type of vagueness exhibited in the sizing constraints given by designers. Overall, it enables a gradual refinement modelling approach reflecting the minimum commitment modelling principle. Our research so far indicates that intervals provide a suitable underlying representation for the size approximation exhibited in early design information. Extension of the representation may be investigated to model a designer's preference to certain values over a range by, e.g., incorporating other known uncertainty handling techniques such as probability. Such preferences may be important since they reflect the designer's experience acquired over years of practice about a specific subject. In the context of the entire representation framework presented in the paper, a powerful interval constraint solving or satisfaction engine is required to support the extension of the interval representation to cope with dependent size constraints of inequality and/or equality types.

With respect to location approximation, at this stage of our study, we feel that the notion of uncertain region is very useful and enables the gradual refinement approach. However, it is insufficient to represent an uncertain region merely as three intervals, as in the current work, which represent the physical extent or scope of the region along the three co-ordinate axis directions. This representation simplifies the shape of the uncertain region as a cuboid. As the design proceeds and more complex shapes are introduced into a configuration, such a simplification limits the system's capacity in modelling compact