A System for Constructing Boundary Representation Solid Models from a Two-Dimensional Sketch— Topology of Hidden Parts

P. A. C. Varley, R. R. Martin

Dept. of Computer Science
Cardiff University
Cardiff, Wales, UK
{Peter.Varley, Ralph.Martin}@cs.cf.ac.uk

Abstract

This paper discusses research results useful for producing a system which converts a two-dimensional computer sketch of a single homogeneous polyhedral object into a boundary representation solid model. An overview of our approach to such a system and its main algorithms is presented elsewhere.

This paper gives more detail of our method for creating the topology of hidden parts, given a sketch for which the geometry has been given a preliminary interpretation, and the object has been assigned to one of several regularity categories. It discusses various alternative methods which were considered, and explains which were chosen and why. The topics covered in detail are the choice of method, the types of hypotheses which can be made about the hidden topology, and the special case processing used for particular categories of object.

1. Introduction

This paper discusses research results useful for producing a system which converts a two-dimensional computer sketch of a single homogeneous object into a boundary representation solid model. The utility of such a system has been argued elsewhere [8, 11].

The problem is to convert the sketch into a boundary representation solid model of the most plausible 3D interpretion of the sketch, and to do so in a reasonable time (a second or less on a powerful personal computer is "reasonable").

Although the theoretical impossibility of perfect conversion of 2D to 3D is both obvious and well-known, there exists an expanding set of objects for which conversion can be achieved in practice. Some assumptions concerning the sketch are required. We assume that the sketch is of a single manifold polyhedral object. Topologically, we assume that all vertices in the object are trihedral, that no face contains more than one loop, and that the sketched object contains no through holes. Geometrically, we assume that the object is in general position, with no faces lying perpendicular to the view direction. Lastly, we assume that the object has been sketched from the "most informative viewpoint", i.e. that there is nothing at the rear of the object which could not reasonably be inferred from the visible part of the object.

In terminology, we use *region*, *line* and *junction* to denote *atoms* of a two-dimensional *sketch*, and *face*, *edge* and *vertex* to denote atoms of the three-dimensional *object* which has been sketched. A junction where two lines meet is *biconnected*; a junction of three lines is *triconnected*.

We also distinguish *recognition*, where the computer chooses one of a finite set of candidate objects, from *interpretation*, where the computer creates an object based on a set of assumptions which allows a potentially infinite set of constructible objects. We are interested only in the latter.

Several approaches to this problem are possible, and have been summarised elsewhere [11]. Of these, we have chosen Grimstead's system [6] as our reference point, as this makes the same assumptions as our system and comes closest to achieving our aims. It produces valid 3D interpretations of many classes of valid sketches, it produces the most plausible interpretations for a subset of these, and it is quick enough to be considered interactive.

Grimstead's system is analysed in detail elsewhere [11]. One problem with it is that by generating the geometry of the object using a linear system of equations and finding the best fit, the system does not necessarily enforce individual constraints. For example, faces may end up nearly but not quite parallel, and corners may end up nearly but not quite perpendicular, when enforcing parallelism and perpendicularity would produce a more plausible and aesthetically appealing object.

Another aid in producing appealing geometry is the enforcement of constraints based on symmetries and regularities [9]. We use the terminology that rotation and reflection are *symmetry operations*. Rotation axes and mirror planes are *symmetry elements*. Other artefacts which give clues to the structure of an object, such as parallelism or extrusion, which do not meet the strict definition of symmetry, are *regularities*.

Elsewhere [11], we have described a method which improves on Grimstead's system by interpreting a larger subset of valid sketches and by producing more plausible topological completion and geometric details of the resulting object. Briefly, this process can be subdivided into three main steps: preliminary processing (including frontal geometry and sketch categorisation), topological reconstruction and geometrical finishing.

The aim of preliminary processing [12] is to deduce as much information as possible concerning the sketch, without making changes or additions: lines and junctions are labelled, candidate parallel line pairs are identified, a preliminary *frontal geometry* (the 3D geometry of the part of the object visible in the sketch) is produced, potential local symmetries are identified, and an attempt is made to categorise the object portrayed in the sketch as one or more of a number of cases which simplify reconstruction.

Figures of merit are assessed for hypotheses and used to adjudicate between competing hypotheses, and are justified and described in more detail in [12]. Numerically, they may be *multiplied*, $F_{A \cap B} = F_A \times F_B$, reducing the merit, or *reinforced*, $F_{A \cup B} = 1 - (1 - F_A) \times (1 - F_B)$, increasing the merit.

The aim of topological reconstruction (described in this paper) is to produce a complete topology for the object. From the frontal geometry [12], the list of symmetries and regularities, and the categorisation, various hypotheses are made concerning the topology of the hidden parts. Repeatedly, until a topologically-complete object is produced, the best such hypothesis is chosen according to an estimate of merit, and the corresponding hidden part is added to the object.

Geometric finishing [13] attempts to produce the best geometrical interpretation of the recovered topology given the list of symmetries and regularities. Identification of a symmetry element or regularity in an object produces one or more *constraints*, which limit the possible positions of the faces, edges and vertices. The system attempts to satisfy as many of these constraints as possible, in descending order of merit.

The rest of this paper gives technical details of how certain key alternatives in creation of the topology of hidden parts were chosen, and also includes progress since [11]. We conclude by outlining plans for future work in this area.

2 General-Case Recovery of Topology of Hidden Parts

Previous stages in our system [12] make inferences concerning the visible part of a sketch: junctions and lines have been labelled, inferences have been made about local symmetries and regularities and the possible categories to which the object as a whole may be allocated, each such hypothesis is given a figure of merit, and a preliminary estimate has been made of the visible 3D geometry. We now wish to construct a topology (a complete and consistent face/edge/vertex structure) which includes the hidden part of the object.

Seeking the most plausible topology can be viewed as a search through the tree of possible topologies. We therefore define: a control mechanism for our search; the nature of a branch; and the means by which branches are generated.

2.1 Control Mechanism

There are several possibilities for the control mechanism, including:

A- Recognition of known objects

B- Generation of all reasonable topological completions

C- Reconstruction based on classification

D- Greedy method with fixed list of choices

E- Greedy method with merit-based choices

F- Backtracking

2.1.1 Recognition of Known Objects

Several systems exist for choosing an object from a database of known objects given an input sketch ([10] is a recent example). This is not our objective. However, since it is possible that there will be common objects which defeat whatever method we choose, for pragmatic reasons we may choose to recognise these particular objects as special cases, extract their completed topological form from a database and adjust the geometry to match the sketch.

2.1.2 Generation of All Reasonable Topological Completions

For completeness, we mention the idea of generating all "reasonable" topological completions and then picking the best as determined by assessing a figure of merit for each. We surmise that even if a cutoff is implemented such that only completions with no more hidden faces than visible faces are generated, this approach would take several minutes even for relatively simple sketches. This is not compatible with the desire for an interactive system.

2.1.3 Reconstruction based on Classification

Given that most objects can be categorised, as described in fuller detail in [12], it is possible to attempt to use different methods for reconstructing the topology depending on the object categorisation. Rather than search the entire tree, we follow the one branch which leads to our preferred destination. For example, the topology of an extrusion can be completed simply from the visible end-cap.

There are two obvious problems with this approach:

- several combinations of categories may simultaneously be valid — as the number of special categories increases, there may be a combinatorial explosion and writing special-case code for each valid combination will then be impractical

- some sketches resist categorisation entirely — there must therefore be a general-case method to interpret these

Ideally, we would wish to aim for our general-case method to handle as wide a variety of sketches as possible, minimising the need for special-case methods. However, special case methods may be faster and more robust, and may thus be a practical necessity in a realistic system.

2.1.4 Greedy Method with Fixed List of Choices

Straightforward greedy methods have been used successfully in simple systems. As an example of these, we consider a process with a list of methods for making a sequence of "moves" towards completing the object. The list is ordered, such that if a move of type M_1 is possible it is made; thus a move of type M_n is made if and only if no moves of types M_1 to M_{n-1} are possible in the current state of the partially-completed object. After each move, the partially-completed object should be "more complete" (or at least no less complete) by some measurable criterion.

We have tried various deterministic lists of moves ordered in this way. For at least the moves we considered, we found no ordering which works for *all* of our test cases, and it seems likely that human ingenuity in devising counter-examples will prove sufficient to defeat any which are proposed.

To illustrate this, consider the objects in Figures 1 and 2 respectively. Two heuristics are used. T-junction completion extends the partial line through the T-junction to a hidden vertex, which is connected by hidden edges to the next visible edge around the partial face, and to the next L-junction around the background. Orthogonal vertex completion connects two or three existing L-junctions with missing edges along differing axes by adding a hidden vertex and connecting it to the L-junctions using hidden edges.

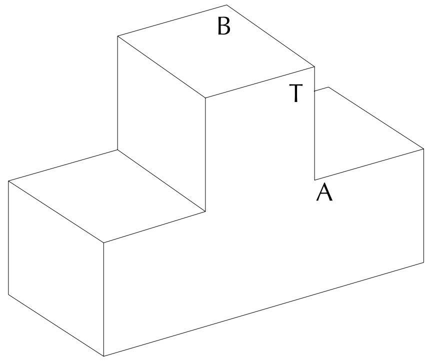

Figure 1. T Block

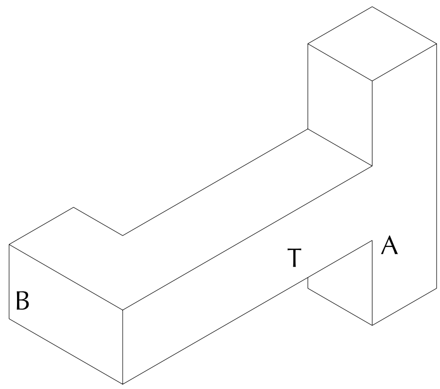

Figure 2. J Block

If T-junction completion takes priority, the T-block is completed successfully: after T-junction completion, there are only three remaining L-junctions, and these can be linked by adding a single hidden vertex. However, if orthogonal vertex completion takes priority, one of the vertices required for T-junction completion can be used mistakenly (see Figure 3). Considering the J-block,

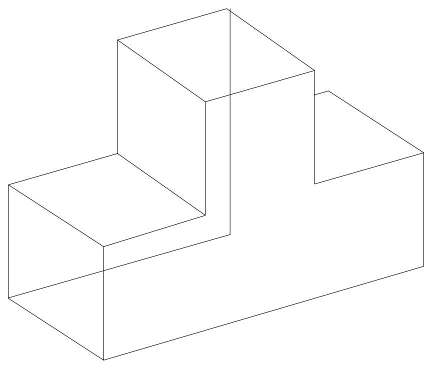

Figure 3. T Block Error

if orthogonal vertex completion takes priority, Figure 2 can be completed successfully. It is only if T-junction completion takes priority that it fails: the background vertex to which the new hidden vertex should be connected does not yet exist (see Figure 4).

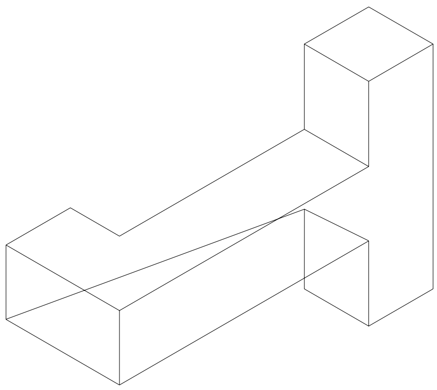

Figure 4. J Block Error

2.1.5 Greedy Method with Merit-Based Choices

If the choice of move is not determined by generating them in a predefined order and choosing the first valid one, evidently some other method is required in order to choose between them; this can be generalised as "merit-based". If possible, we should attempt to combine the advantages of the greedy method with those of maintaining "tentative hypotheses" (in considering interpretation systems, Hinton [7] argues in favour of tentative hypotheses, where a number of competing hypotheses are maintained until a choice needs to be made). While the object remains incomplete, various possible moves towards completion are suggested on the basis of surviving tentative hypotheses. The various moves are assessed on the basis of figures of merit [12], the highest-rated is chosen, and the tentative hypotheses are re-assessed in case they are invalidated by the newly-made move. As with fixed-list methods, each move is designed to leave the partially-completed object nearer to, or at worst no further from, completion by some measurable criterion. Two variants of this are described in more detail in Section 2.2 below.

2.1.6 Backtracking

With the greedy methods described above, it is possible that the process may reach a state where it can be identified that it is no longer possible to reconstruct a valid object — for example, if the object contains only two incomplete junctions and there is already an edge linking them, no number of further additions can create a valid object, and something must be removed. In order to guarantee that the process always produces a valid object from a valid sketch, a backtracking mechanism is required.

Inclusion of backtracking could overcome many of the cases where the greedy methods fail. However, backtracking slows the system down, sometimes unacceptably so, and therefore we only wish to invoke it if the system fails to produce a valid object. It is more common for the system to produce an incorrect topological completion (one which is valid, but not the one expected by the user) — this would not cause backtracking to be invoked, as the system would be unaware that anything was wrong. Backtracking could then be invoked manually by the user, but this is undesirable in our opinion.

In implementing the greedy methods, to avoid the potential for a tree growing indefinitely, hypothesis formation is abandoned if, while the object remains incomplete, there are already more hidden faces than visible ones. If this occurs, the completion is deemed to be in violation of the assumption that the object is drawn from its most informative viewpoint, the current topological completion is rejected, and the system backtracks to the last state at which there was a plausible alternative.

A backtracking mechanism also allows a chosen move to be rejected either immediately, if after making the move the object is invalid (see section 2.5 for some examples of this), or if after following all branches of the resulting tree of moves, none of these results in a valid object. The former has been found useful, but the latter is only sporadically useful: in practice, a valid but poor move leads more often to a valid but unexpected object than to an invalid object.

2.1.7 Recommendations

Generation of all reasonable topological completions, by whatever means, is too slow (we estimate by a factor of 1000 for useful objects), and can be rejected until such time as desktop computers increase in speed by this factor.

Since the system must be able to attempt a reconstruction of arbitrary sketches which fall into no predefined category, it must include something resembling either the iterative or greedy methods described above. Although backtracking is easily-implemented and we recommend its inclusion, it is undesirable for an interactive system to *rely* on backtracking, so we prefer the merit-based greedy method.

The ideas of incorporating recognition of specific known objects and reconstruction based on classification for specific special cases of object is considered further in Section 3.

2.2 Types of Move

The methods listed above which are capable of reconstructing arbitrary hypotheses involved making "moves" towards completing the object. The tree being searched is thus one of moves. To identify the nature of each branch, we must consider the various possibilities for the types of move. These include:

- Creation of a new vertex and associated elements
- Creation of a new face and associated elements
- Creation of a new edge and associated elements
- Creation of any new atom (vertex/edge/face) and associated elements
- Pre-interpreted sketch fragments
- Reconstruction from a symmetry element

2.2.1 Creation of a New Vertex

One straightforward possibility for moving towards completion is that a new vertex is added, together with two or three edges to join it to incomplete vertices in the existing object. This can be shown to be a move towards completion: ignoring T-junctions, if a biconnected hidden vertex is added, two edges are added, so two previously biconnected vertices are completed, and the number of incomplete vertices is reduced by one; if a triconnected hidden vertex is added, three edges are added, so three previously biconnected vertices are completed, and the number of incomplete vertices is reduced by three.

2.2.2 Creation of a New Face

Another possible move is to add a new face with at least two vertices and at least one edge which are existing parts of the object. This is more flexible than adding a new vertex, but the criterion of moving towards completeness is less easy to define. For example, although adding a quadrilateral face containing three existing vertices (two of them biconnected) decrements the number of incomplete vertices, adding a pentagonal face containing three existing vertices has no effect on the number of incomplete vertices and adding a hexagonal face actually increases it.

2.2.3 Creation of a New Edge

Some alternative move types potentially leave the partially-recovered object in a state which is not equivalent to a frontal-geometry object, or which leaves the object less complete than it was before the move was made. One example is the creation of a new edge, completing a junction and leading away from it. This is felt to contradict the requirement that each move makes the object more complete — dangling edges (such as those which create T-junctions in the 2D sketch) are a measure of incompleteness, and increasing their number is to be discouraged. Creation of a dangling edge also leaves the partial object in a state which is not exactly equivalent to a frontal-geometry object in that while a T-junction is created at the point where one visible face occludes another face oriented towards the user, no such assumption can be made concerning an arbitrary hypothesised hidden edge.

Creation of a single hidden edge joining two incomplete vertices suffers from neither of these problems — it is clearly a move towards completion, reducing the number of incomplete vertices.

2.2.4 Creation of Any Type of Atom

There are disadvantages to creating a tree where the branches are of different move types. The major problem occurs when the attempt is made to combine hypotheses which say the same thing (see Section 2.5) — this is straightforward if only face-based moves are generated, and is in practice very useful if vertex-based moves are used.

However, if vertex-based moves are the norm, there may be circumstances where two existing incomplete vertices must be joined by new single edge; it is therefore necessary to allow mixed move types to this extent.

2.2.5 Pre-interpreted Sketch Fragments

Draper [4] suggests making use of pre-interpreted picture fragments in sketch reconstruction — for example, in the case of the T-piece problem illustrated below in Figure 12, the correct solution could be hypothesised as a single move. We have not investigated this possibility yet. Depending on how common we require the sketch fragments to be to justify special-case processing, the number of common sketch fragments could either be impracticably large or so small as to make the idea pointless.

2.2.6 Reconstruction from a Symmetry Element

Complete reconstruction from a symmetry element is discussed below, in the section on special-case reconstruction.

There are some sketch artefacts from which part but not all of the object can be deduced. For example, by pairing atoms in the sketch with their equivalents after reflection across a mirror plane we can deduce from those atoms with no pairings the presence of many atoms in the hidden part of an object; however, this method cannot be used directly to deduce hidden faces or edges bisected by the mirror plane.

2.3 Suggested Move Strategy

We require a strategy which is in principle capable of general-case construction of arbitrary objects. The choices are: limit moves to face-based moves only; limit moves to vertex-based moves and edges joining existing vertices; allow a mixture of face-based, edge-based and vertex-based moves; and allow a more complex mixture of moves. We reject the mixture strategies as overly complex, and compare a strategy using only face-based moves with a strategy using vertex-based moves and edges joining existing vertices. Both strategies have been implemented.

It has been found that a strategy using only face-based moves is capable of completing a wider selection of objects. To illustrate this, consider the front geometry of a dodecahedron. Face-based moves hypothesising pentagonal faces containing three existing occluding junctions can be generated by considering C_5 face rotations and mirror symmetries, and these are adequate for recovering the hidden topology. However, vertex-based moves fail altogether, in that no reasonable hypothesis predicts a single hidden vertex connected to two existing visible vertices.

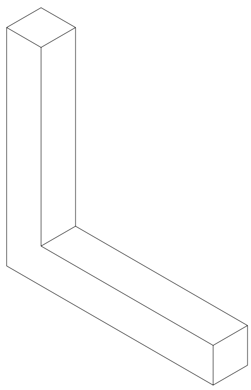

Figure 5. L Block

However, face-based moves are more sensitive to the numerical figures of merit assigned to the various types of move. To illustrate this, consider Figure 5. All reasonable hypotheses for the first move amount to the same thing — a single hidden vertex is connected to the three visible L-junctions. Using vertex-based moves, these hypotheses can thus be merged, and the figures of merit become irrelevant. However, with face-based moves, there are three distinct reasonable hypotheses, each producing one of the hidden faces. With such a simple sketch, this is of no practical importance — all three hypotheses will eventually be used. With more complex sketches, this becomes important, as plausible but wrong hypotheses can be made.

Vertex-based moves will generally recover the hidden vertices of Figure 6 correctly — the several good hypotheses reinforce one another, as they did for Figure 5, to predict the partial completion shown in Figure 7.

However, with face-based moves, the three good hypotheses for the two back faces and the underneath are now competing not only with one another but with alternative hypotheses such as that of a quadrilateral face underneath the wing as shown in Figure 8. Which of these hypotheses wins depends on the figures of merit. It can also be noted that even if the bad hypothesis is accepted, it is possible to recover a valid axis-aligned object, albeit not the one expected. Since a valid object is obtained, backtracking will not be invoked, and so it becomes important to choose figures of merit which give priority to the good hypotheses in preference to the bad ones.

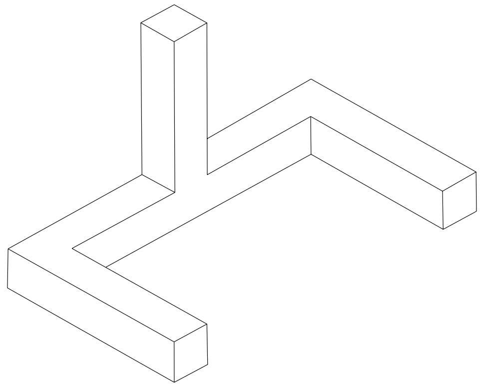

Figure 6. Double Z Block

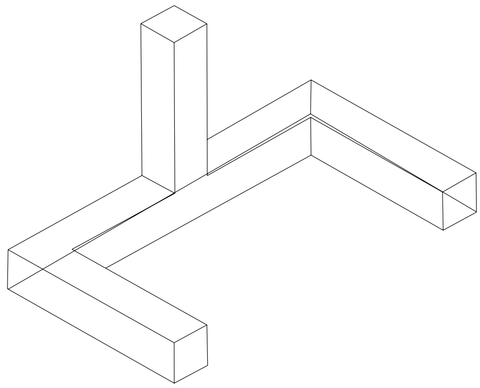

Figure 7. Double Z Block — Right

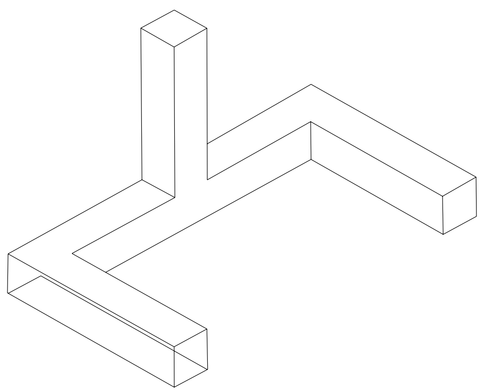

Figure 8. Double Z Block — Wrong

It can be noted that some hypotheses are easier to implement with vertex-based moves, and others with face-based moves. Our subjective view is that hypotheses based on vertex configurations or on extending lines from incomplete vertices are easier to implement using vertex-based moves, while hypotheses based on symmetry operations are easier to implement using face-based moves.

Our preferred stragegy therefore depends on the type of object sketched. For the general case, lacking both axis-alignment and symmetry, we prefer first to try the less sensitive vertex-based version of the general-case method, adding a single vertex at a time in the hope that some sort of regularity will become evident, and only if this fails to produce a complete topology do we use the more generally-applicable face-based method. For sketches with multiple symmetries, and in particular for sketches with few or no visible quadrilaterals, we prefer face-based moves. For axis-aligned sketches, we prefer a variant of the vertex-based moves described below in Section 3.2.

2.4 Types of Hypothesis

In order to generate moves, whether face-based or vertex-based, we must make hypotheses concerning the unknown part of the object based on what we already know. Each hypothesis may generate one or more moves (inappropriate hypotheses may even generate none). We have implemented several types of hypothesis, as described in the following sections.

When hypothesising the presence of a new vertex, its geometry is estimated — this affects the merits of the hypothesis, as described in Section 2.5.

2.4.1 Complete Hypothesis

If any object contains no biconnected junctions and no T-junctions, it is assumed to be complete in terms of vertices and edges — the only things which might need adding are extra faces. These can be detected using the next stage of our system, the loop-detection algorithm described in Section 4. Even if face-based moves are used, loop-detection is simpler and quicker than using the hypothesis/figure-of-merit method, although that too would identify faces correctly (consider, for example, the L-block case described above, after two of the three rear faces have been added — all sensible hypotheses would predict the remaining rear face, but this could be found far more simply by looking for a loop of unused half-edges).

2.4.2 Two Biconnected Junctions

In any object which contains exactly two biconnected junctions and these junctions are not connected by an existing edge, it is reasonable to add a single hidden edge connecting them to complete the vertex/edge framework. In trihedral objects, this can be considered to take priority over any other possible move — the move is made immediately, other hypotheses are bypassed, and the next stage of our system, loop detection, will then identify any remaining faces.

The same deduction can be made in any object which contains exactly one biconnected junction and exactly one T junction, and the biconnected junction is located approximately on a continuation of the line defining the T junction. It is reasonable to extend this line to the biconnected junction to complete the vertex/edge framework.

2.4.3 Mirror Hypotheses

Earlier stages in our system [12] produce a list of mirror chains — each mirror chain tracks a potential plane of mirror symmetry across one or more visible faces. For each such chain which has been identified, an attempt is made to create a list of correspondences between the current partially-complete object and the the mirror-image which would be generated by reflection. Each atom (vertex, edge or face) in the mirror chain is matched to its equivalent after the reflection operation, and then an attempt is made to propagate the matches through the existing part of the partially-completed object. We are seeking unmatched atoms — if these are found, we can then hypothesise moves which add these atoms to the object.

Propagation continues while either a pairing for an unmatched face or a pairing for an unmatched edge can be found. An unmatched edge can be paired to a reflected edge if the reflected edge is not already paired, and either (a) the two vertices of the unmatched edge are paired to the two vertices of the reflected edge or (b) one of the vertices of the unmatched edge is paired to one of the vertices of the reflected edge, and each of these vertices has only one unpaired edge (these criteria allow the pairing of partial as well as complete edges). An unmatched face can be paired to a reflected face if the reflected face is not already paired, they have (or might have) the same number of sides (for partial faces, we know the minimum, not the actual number of sides), all paired vertices and edges on the faces are paired correctly, and there are at least three such pairings.

A further inference can be drawn — if two non-occluding edges are paired, and one face from each edge is paired, the other faces on each edge can also be paired. Omission of this inference is usually unimportant but was the cause of the failure, reported in an earlier paper [11], to interpret Figure 9 correctly. (In this example, pairing across the mirror plane pairs face M with itself and edge A with edge B; applying the inference allows us to pair face N and face P.)

Atoms in the mirror-image which have no correspondence to atoms in the original are assumed to correspond

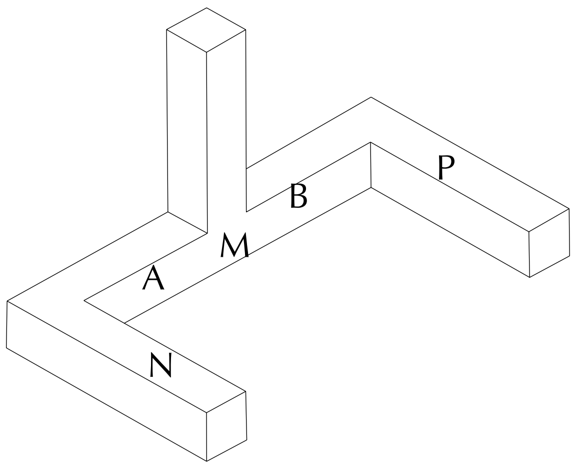

Figure 9. Double Z Block

to extra atoms which will be required, and the appropriate moves for creating them can be generated.

The provisional coordinates of newly-created vertices are determined from the coordinates of the unmatched vertices and the equation of the mirror plane. We have tried various methods for locating the mirror plane:

- for a mirror chain which includes two or more faces, we prefer to use a linear system to find the best solution to a set of equations which place vertices and edge bisectors in the mirror plane; this is our preferred method, but cannot be used for single-face mirrors, where there are only two such points and thus not enough equations; care must also be taken to include each point only once, as our linear-system-solver [1] can be disrupted by redundant equations

- to use a linear system of equations which place in the mirror plane not only vertices and edge bisectors, but also further points constructed by translating the original points by a fixed amount along the face normal; the best-fit plane to these non-collinear points can be found, and this is our preferred method for single-face mirror planes;

- to generate the plane perpendicular to a face, with plane normal along the bisected edge; this is inherently less accurate than the preceding methods, as it relies on the preliminary geometry of a single edge; it cannot be used for vertex-to-vertex mirror planes

- to generate the plane defined by the mirror lines crossing any two consecutive faces in the chain (i.e. the mirror plane normal is the cross-product of the two line vectors); this cannot be used for single-face mirrors, and except for mirror chains of exactly two faces, it does not make use of all lines in the chain so is inherently less accurate

- generate all the correspondences between visible vertices, and use a linear system of equations which place the correspondence bisectors on the mirror plane; this is somewhat slower, and although an improvement in geometry might be expected since more points are being used, no improvement is noticeable in practice

The base figure of merit for mirror hypotheses is the figure of merit for the mirror chain, which is calculated at an earlier stage [12] and is based on how well the hypothesis fits the geometry and a number of topological factors (it increases with the number of faces in the mirror chain, and also increases if the chain starts or ends at an object boundary). If the object is categorised as semi-axis-aligned with mirror plane and this is the chosen mirror chain, the figure is *reinforced* by the merit for this categorisation. If, when propagated, the pairings pair incompatible line types, the merit is halved (the resulting topological hypotheses provide useful local information even if no consistent global solution). The merit is divided by the number of new atoms (vertices or faces) being hypothesised – the more new atoms are required, the less reliable the mirror chain.

As mirror chains are better at generating topology than geometry, the figure of merit for hypothesised vertex geometry is lower than that for the hypothesised topology (we use a fixed multiplier of 0.1).

2.4.4 Rotation Hypotheses

Hypotheses derived from rotational symmetries use a similar algorithm, deriving equivalent vertices, edges and faces before and after the symmetry operation. In the case of rotations about a face centre, we seed this process by noting that the face is its own equivalent, and storing the equivalent vertices and edges before and after the operation for each vertex and edge on the face; the axis of rotation is calculated by finding the best plane through the vertices on the face, and extending a normal to this plane from a point in the centre of the face. In the case of C_2 rotations about an edge midpoint, we seed the equivalence process by noting that the edge is its own equivalent, the vertices at either end of the edge are equivalent to one another, and the faces adjacent to the edge are equivalent to one another; the axis of rotation is as close as possible to the sum of the face normals of the adjacent faces while being constrained to be perpendicular to the edge direction. In the case of C_3 rotations about vertices, we seed the equivalence process by noting that the vertex is its own equivalent, and storing the equivalence relations of the edges and faces adjacent to the vertex; the axis of rotation is the sum of the face normals of the adjacent faces.

Equivalence is propagated through the object in a similar way to mirror pairing propagation as described above, the only difference being that we must use the appropriate symmetry group (e.g. for C_3, we know that if $A_1 \mapsto A_2$ and $A_2 \mapsto A_3$ then $A_3 \mapsto A_1$).

The base figure of merit for a hypothesis based on rotational symmetry is the figure of merit for the symmetry element. As with mirror planes, this is halved if the pairings pair incompatible line types, and divided by the number of hypotheses begin generated by the individual element.

Rotations about edge centres are included for completeness, and it may be better to omit reasoning based on them from a practical system: there are few if any sketches for which these give any additional topological information (for example, the two hidden faces of Figure 10 which can be deduced from edge-centred rotation hypotheses can also be deduced by other means), and the calculation which produces geometric position estimates for hidden vertices is relatively slow and comparatively inaccurate.

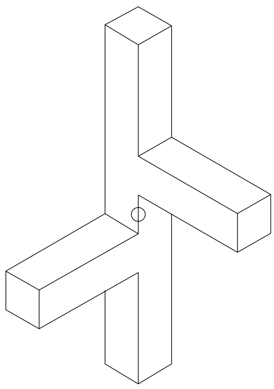

Figure 10. Edge-Centred Rotation

2.4.5 Axis-Alignment Hypotheses

If the preclassifier detects that the object might be orthogonal, 2D lines are extended from incomplete vertices along the remaining axis directions in the same manner as in Grimstead's system. Vertices are hypothesised where these lines intersect, and moves generated which would create these vertices. (The predicted x and y coordinates of the hypothesised vertex are the coordinates of the intersection; the z coordinate is predicted by taking the mean of values obtained by extending 3D lines along the axes from the 3D starting-points of the 2D lines.)

The base figure of merit for an axis-aligned hypothesis is the figure of merit for the categorisation as axis-aligned. This is divided by the number of crossings of each of the lines which produced the hypothesis. If three of the proposed lines intersect at approximately the same point, the three merit figures are reinforced.

The divisor is intended to bias selection in order that unarguable moves are made first — if a line crosses only one other line, then the likelihood that there is a vertex where they intersect is high; if it crosses several other lines, the one vertex somewhere along the line could be at any of the intersection points. We can generate crossing numbers (the number of other lines crossed) for each line. Dividing by the lower of the two crossing numbers seems intuitively more plausible — if only one line crosses line A then it seems natural to place a vertex there — but can lead to problems in cases where the true lines crossing line A come from vertices which have not yet been created; our experience is that dividing by both crossing numbers gives better results in practice.

In deciding on the direction in which to extend the line, we use the mean direction for the bundle so far unused by the vertex. For lines extended through T-junctions, there is an alternative possibility: we could extend the line from the visible vertex through the position of the T-junction as estimated during the preliminary geometry stage [12]; this produces a much less accurate estimate and has been rejected. Since the resulting geometry is relatively reliable, the figure of merit for the geometry is greater than that for the topology of the hypothesis (we use $\sqrt[n]{(F_T)}$, where n is the number of lines crossing at a point, 2 or 3, and F_T is the figure of merit for the topological move).

Currently, we do not reduce the figure of merit if the two estimates of a hidden vertex z coordinate are significantly different, or if the predicted z coordinate would make the vertex visible, as initially we considered that either of these conditions would be more likely to result from faults in the preliminary geometry rather than faults in the hypothesis. This decision may be reconsidered as more data become available.

Additionally, if two of the projected lines are almost collinear, it is probable that they are really the same edge, and the two originating vertices should be connected. The figure of merit is high, decreasing as the angle between them increases and the distance between them increases. We use a base value of the figure of merit for the categorisation as axis-aligned, multiplied by the figure of merit for parallelism [12] between the two lines and by $(100 - D_{AB})/100$, where D_{AB} is the distance (in pixels) between vertex B and the line extended from vertex A (a practical system should use a proportion of the sketch size rather than an absolute value; we aim for sketch sizes around 800×800 pixels). As with line crossings, the decision not to reduce the merit if the depth coordinates of the two lines differ significantly may be reconsidered in the light of further experience.

2.4.6 Semi-Axis-Aligned Hypotheses

Similar hypotheses to those for axis-aligned sketches, though with a lower figure of merit, are produced for semi-axis-aligned sketches. Lines are extended from each existing vertex in the directions of those of the three axes not used so far by the vertex, and the positions of hidden vertices hypothesised as above. The base figure of merit is the reinforced figure of merit for the two semi-axis-aligned categories (with and without mirror plane); in the case of non-axis-aligned biconnected vertices, two or three lines will be projected (along the axis directions not used so far by the vertex) but since there will be only one further true edge added to the vertex, only one of these projected lines generates correct hypotheses. Thus moves based on lines projected from such a vertex have their figures of merit divided by 2 or 3 as appropriate.

2.4.7 Extrusion and Frustum Hypotheses

As noted below in section 3.1, we now prefer special-case completion of extrusion and frustum topology. Previous versions of our system generated moves based on this categorisation, using the categorisation figure of merit for the moves, which would then reinforce or compete against other hypothesised moves. It is straightforward to generate moves, either for the vertex/edge approach or for the face approach, based on the assumption that the sketch is of an extrusion or a frustum. In the former case, for each vertex on the hidden end cap not visible in the sketch, a move creating the vertex can be hypothesised. In the latter case, the hidden end cap and any invisible quadrilateral side faces can be hypothesised.

2.4.8 Uniformity Hypotheses

No additional hypotheses are generated for other special cases such as regular solids, as it has been found that the mirror and rotation hypotheses described above are adequate for completing such an object. Moreover, we note below in Section 3.4 that we prefer special-case completion for such sketches.

2.4.9 Local Configuration Hypotheses

We note that quadrilateral faces occur frequently in engineering practice. By hypothesising moves which create or imply quadrilateral faces, we can bias the system towards quadrilateral construction by reinforcing any corresponding existing moves, and produce quadrilateral-containing interpretations of some difficult (asymmetrical and irregular) sketches where none of the above methods hypothesises moves. Two local configurations of vertices lead to useful hypotheses.

Firstly, given any two biconnected vertices A and B, if they are separated by a single triconnected vertex C then adding a new vertex D to form a parallelogram $ACBD$ is plausible. The figure of merit is based on a fixed value for this type of move (we use 0.25) multiplied by the proportion of quadrilateral faces in the partially-completed object and the figure of merit for the object being axis-aligned or semi-axis-aligned.

If they are separated by two triconnected vertices C and E, then adding an edge to join A and B to form the quadrilateral $ACEB$ is plausible. The figure of merit is based on a fixed value for this type of move (we use 0.09) multiplied by the proportion of quadrilateral faces in the partially-completed object, the figure of merit for the object being axis-aligned or semi-axis-aligned, and the figure of merit for 3D parallelism of the supposedly parallel lines AC and BD.

2.4.10 Three Biconnected Junctions

If there are exactly three junctions remaining with only two edges apiece, it can be hypothesised that there exists a single hidden vertex connected to all three. This generates one hypothesis for vertex-based moves and three for face-based moves. The figure of merit is fixed and high (we use 0.7). Since in practice this hypothesis, if made, is always accepted, it might be preferable to include it amongst the hypotheses which are accepted automatically.

2.4.11 T-Junction Hypotheses

Hypotheses can be made for T-junction completion. A true vertex must exist somewhere further along the line past the point at which the T-junction intersects the occluding line. It is plausible that this vertex is directly connected to the next real vertex (vertices) obtained by following the occluded region (regions) for arrow (concave, convex) T-junctions around along the occluding line (this would join vertices T to vertices A in Figures 1 and 2). The figure of merit is a constant value (we use 0.6) multiplied by the proportion of known faces in the object which would have the same number of sides as this method would predict. It is also worth generating the hypothesis that for arrow T-junctions ($Tbaa$ as in Figure 2 or $Tbab$ as in Figure 1), the true vertex is connected to the next incomplete junction in the other direction around the object boundary (this would join vertices T to vertices B in these two Figures). This may or may not be true in practice, so the figure of merit is lower (we use 1/3 of the figure of merit of the more reliable T-hypothesis).

2.5 Hypothesis Adjudication

Except for the three "complete" cases, where the vertex/edge framework is (i) already complete or will be after (ii) adding a single edge linking two vertices or (iii) extending a line through a T-junction to the remaining biconnected vertex, moves are selected on the basis of generating candidates by the methods described below and picking the one with the best figure of merit.

Where two or more methods suggest the same move, the figure of merit for that move is *reinforced*. Additionally, where two candidate moves appear to represent the same thing (the details of this vary depending on the type of move implemented), the hypotheses are merged into one and the figure of merit *reinforced*.

As described above, whenever a new hidden vertex is hypothesised, a prediction is made of its geometric position, and this geometric prediction has its own figure of merit F_G. Whenever equivalent hypotheses are merged, the predicted positions are merged by calculating a weighted mean position, the weights being the geometric figures of merit.

Any hypothesis which places a new vertex in a position in which it would be visible given the existing faces has its figures of merit reduced, ($F'_T = F_T/10; F'_G = F_G/100$). This is intended to allow for the fact that the geometry of the partial object is at best provisional, and thus locations derived from it are inaccurate, without rejecting better hypotheses which place the new vertex in a hidden location.

Figures of merit are also reduced if the hypothesised move conflicts with beliefs about the object as a whole, or for other reasons which reduce their plausibility:

- if the figure of merit for the complete object being axis-aligned is non-zero, any move hypothesising an odd-sided face is downgraded ($F'_T = F_T \times (1 - F_{\text{axis-aligned}})$);
- if the figure of merit for the complete object being an extrusion is non-zero, any move hypothesising faces which are neither quadrilaterals nor compatible with the invisible end cap is downgraded ($F'_T = F_T \times (1 - F_{\text{extrusion}})$);
- if the figure of merit for the complete object being uniform (regular, semi-regular or prismatic) is non-zero, any move hypothesising a face which is not compatible with a visible face is downgraded by multiplying by the figure of merit for the object not being uniform ($F'_T = F_T \times (1 - F_{\text{uniform}})$);
- two or more hypotheses generate moves with the same connectivity but with incompatible edge types — choice between them should be deferred until a later iteration (by which time one or other of the hypotheses may have been abandoned) providing there are other reasonable moves; ($F'_T = F_T/2.5$);
- an existing triconnected vertex is to be further connected to a new vertex ($F'_T = F_T/10$) — the choice of divisor is arbitrary, but it should be large enough to ensure that such a move is only accepted if there is no alternative; in principle, the divisors increase in increasing order of seriousness of the problem, with this problem being not quite as serious as the ones that follow — although it contradicts one of our assumptions, this assumption may be relaxed in future;
- an existing vertex in a face-based move is already triconnected but no existing edge joins it to the next vertex around the face; ($F'_T = F_T/13$);
- an existing vertex in a face-based candidate move already lies on three existing faces; ($F'_T = F_T/13$);
- a T-junction extension does not lie approximately along the line defining the T-junction, as can happen as a result of hypothesised vertex positions being merged ($F'_T = F_T/20$);

Figures of merit can sometimes be increased:

- A face-based candidate move which hypothesises no new vertices or edges is reinforced — any such hypotheses is likely to be reasonable at worst, and might be very good ($F'_T = (1 + F_T)/2$)

Selection of the best candidate is straightforward with face-based moves: any candidate moves which create the same face are merged, after which the face with the highest figure of merit is chosen.

A rather more complex method is used for vertex-based moves, to allow for the possibilities of edge-only moves linking existing vertices and to choose between making a new vertex biconnected or triconnected.

- For each existing vertex i in the object, evaluate F_{V_i}, the *reinforced* figure of merit for all candidate moves which create a hidden vertex X linked to vertex V_i
- Pick vertex A, where F_{V_A} is the highest of the evaluated F_{V_i}s.
- From the moves which add a single edge linking two existing vertices, pick candidate move E with the highest figure of merit, F_E
- If $F_E > F_{V_A}$, connect the two vertices directly by adding a single new hidden edge.
- Else ($F_{V_A} \geq F_E$)
    1. Using only those candidate moves which connect a new vertex X to the existing vertex A, evaluate for each existing vertex j other than A the *reinforced* figure of merit F_{V_j} for a connection between X and j
    2. Pick vertices B, C and D, where F_{V_B} is the highest of the evaluated F_{Vj}s, F_{V_C} the second highest, and F_{V_D} the third highest

3. If no vertex C exists, add vertex X and edges connecting it to A and B
4. Else, if any vertex D exists, add vertex X and edges connecting it to A and B (we will choose between edges XC and XD at a later iteration, after re-evaluating the various hypotheses)
5. Else, if $2F_{V_B} > F_{V_A} + F_{V_C}$, add vertex X and edges connecting it to A and B (most of the moves which predict edge XA also predict edge XB but not edge XC)
6. Else, if $F_{V_B} > 2F_{V_C}$, add vertex X and edges connecting it to A and B (we lack confidence in edge XC at this stage)
7. Else, if there are exactly four incomplete vertices in the partially-completed object, add vertex X and edges connecting it to A and B (as adding XC as well will leave the object with only one incomplete vertex, an illegal state)
8. Else, add vertex X and edges connecting it to A, B and C

This method can be simplified for axis-aligned sketches, as noted below.

Hypothesis adjudication stores the hypotheses and their figures of merit, in order that if the top recommendation is rejected, either immediately or later, the next recommendation can be tried instead. Thus the result of making a move can be tested, and the move rejected if the results are unacceptable.

A move is rejected if, while the object remains incomplete, the resulting tree of subsequent moves is empty. It is also rejected if, after making the move, the object becomes invalid or too complex:

- any of the vertices has non-computable coordinates (e.g. $\cos^{-1}(2)$ is non-computable);
- the object contains exactly two biconnected vertices, already joined by an existing edge, and there are no remaining T-junctions;
- the object has been completed as far as possible (all required atoms have been identified), but not all edges lie on two faces;
- the object remains incomplete, but there are already more hidden faces than visible faces (this prunes out long tree-searches which can occur when a poor choice of initial move leads to the system adding more and more detail to the back of the object in an attempt to produce a valid object).

2.6 Problems Encountered

In testing with complex sketches, it was found that most of the problem cases (failure to find a topological completion, finding a bad completion, and taking too long to find a completion are all problems) resulted from one of two causes.

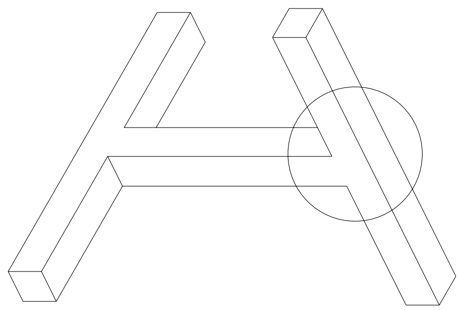

Figure 11. H Block

One occurs where a local feature such as Figure 12 produces the incorrect interpretation (shown on the left) rather than the correct one (shown on the right). This occurs when completing the object in Figure 11, and a similar problem occurred in earlier versions of the testbed with Figure 6. The problem is not so much that the resulting object will be incorrect (the geometry will be almost right, and capable of being "healed " [2]) as that symmetry and regularity artefacts are lost by accepting the poor hypothesis, thus increasing the time taken to find the best completion and the likelihood that a poor completion will be chosen instead.

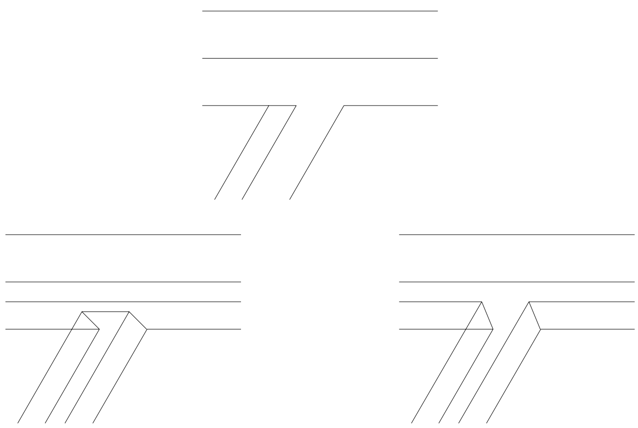

Figure 12. Feature and Interpretations

The other problem occurs with sketches where two T-junctions correspond to the same hidden vertex. Although our system handles this situation correctly when the appropriate move is chosen, the hypotheses which generate these moves tend to have low figures of merit since assumptions based on T-junction extension are always tentative to some extent. Construction of the correct object is thus highly sensitive to the numerical values assigned to figures of merit.

Although it remains likely that further investigation will find methods of assigning figures of merit which handle such cases completely, the fact that such investigation is necessary shows that it is not simple to translate the method described here into a reliable tool which can be shown to handle all cases well. It is therefore worth investigating whether short-cuts can be taken which bypass the hypothesising process, at least for some types of object; the next section discusses these.

3 Special-Case Recovery of Topology of Hidden Parts

While recognition of special cases of objects cannot solve the general problem of reconstructing the topology of sketched objects, it can nevertheless be a useful tool in that many sketched objects belong to special classes, and identifying these may lead to special-purpose methods which are faster, more robust, or both. We now list our recommendations for the special categories we identify. These refer to whole objects. We have, as yet, found no use for pre-interpretation of sketch fragments. As our collection of test sketches expands, this may be reviewed should it be the case that we find common sketch features which are frequently misinterpreted.

3.1 Right Extrusions and Frusta

The topology of an extrusion or frustum is easily reconstructed by creating a back end cap with the same topology as a mirror-image of the front end cap, and joining equivalent vertices in the two end caps by side edges and faces. This is easily implemented and in practice, there are benefits both in terms of speed (special-case topological recovery for extrusions is very quick, and extrusions are common) and in reliability (the general-case hypothesis mechanism is given no chance to make mistakes).

For simplicity, any sketch which has been categorised as an extrusion and as also belonging to some other compatible category, including prisms, cubes and other axis-aligned extrusions, is processed simply as an extrusion.

3.2 Axis-Aligned Objects

For axis-aligned objects, we use a variant of the general-case hypothesis process described above, using vertex-based hypotheses, where the hypotheses are limited to those generated by the crossings of lines extended from incomplete vertices. The other methods of generating hypotheses as listed above are still used, but their effect is to reinforce existing hypotheses, not to add new ones. Although this seems to work well in practice, it seems likely that there are valid axis-aligned sketches for which the hidden vertices cannot be generated by crossing lines; such sketches would thus be processed by rejecting the assumption of axis-alignment and backtracking to restart topological construction. Whether this is adequate is a point for further experimentation.

3.3 Single Symmetry Dominates

The vertex/edge framework of sketches categorised as semi-axis-aligned with mirror plane, and also of any other sketch containing a single high figure-of-merit mirror plane or rotation axis, can often be derived from applying the symmetry operation; the loop detection method (see Section 4) can then be used to find the faces. Even where the resulting framework is not complete in itself (for example, constructing topology from a mirror plane will not create an edge bisected by the mirror plane), using the general method to complete the remainder of the object will be quicker and more reliable than hypothesising new vertices or faces one at a time for all of the hidden part of the object.

3.4 Platonic and Archimedean Solids

We find that the general-case method produces correct topology for all Platonic and Archimedean solids which meet the trihedral assumption, and does so in acceptable interactive time for the dodecahedron but not the truncated icosahedron. However, although the topology of Platonic and Archimedean solids can be completed using the general case, there is no benefit in doing this if (as is generally the case) the geometry must then be reconstructed as a special case. Thus, in a practical system, both the topology and the geometry of a Platonic or Archimedean solid should be read from a database of the known finite set of such solids. (In our system, this is an option, since the regular solids make useful test cases to ensure that objects with multiple high-merit symmetries are handled correctly.)

3.5 Multiple Symmetries

Since (a) symmetry can be used to hypothesise entire faces in a single move, and (b) vertex-based moves can fail for sketches containing no quadrilaterals (and indeed do fail for the dodecahedron), we prefer to use the face-based version of the general-case hypothesiser for sketches which contain multiple high-merit symmetries. Since objects with multiple C_5 / C_6 symmetries are believed to be uncommon in engineering practice, a practical system could probably omit this case. Our only test sketches which meet the criteria of multiple symmetries are the Platonic and Archimedean solids and highly-symmetric axis-aligned solids, both of which are handled as special cases as described above.

4 Remaining Loops and Validation

Once the vertex connectivity has been completed, as described above, any additional faces required to make all vertices trifacial and all edges bifacial are generated by testing for unused loops of half-edges in the object. If this completes the object (we have no test case where it fails) the object is deemed to be topologically complete.

- Repeat, while any vertex is connected to fewer than three faces
  - Choose an unused half-edge
  - Find the smallest loop of unused half-edges including the chosen half-edge (using Dijkstra's algorithm [3])
  - Create a new face using this loop of half-edges

Although all of our test cases which reach this point generate a correct topology, there are some which produce topologies which have no valid non-self-intersecting geometric realisation. We are still investigating what to do about this. It is not sufficient to identify cases where the *provisional* geometry is incorrect — we have a number of test sketches for which topological reconstruction initially produces incorrect geometry (in particular, intersecting faces) but the following geometric finishing stage is able to correct this. Ideally, we require a topological reasoning method which can identify and discard those topologies which cannot be realised geometrically. Any solid meeting our assumptions has genus zero and hence its face/edge/vertex graph can be embedded in a sphere, and is thus a planar graph [5]. In the case of interest, this reduces to Euler's formula, $F + V = E + 2$. We reject topologies which fail to meet this criterion. However, this by itself does not guarantee that it is possible for the realised solid object to have convex and concave edges where we have identified them, or for all faces to be made planar simultaneously. In the absence of a symbolic reasoning method for rejecting topologies which cannot be realised using planar geometry with appropriate convex and concave edges, a substitute approach would be to produce a figure of merit for the topological completion. This could be based on geometrical considerations such as self-intersecting geometry as well as topological considerations such as how well the constructed topology matches predicted symmetry and regularity artefacts. If the figure is below a threshold value, the completion is stored but one or more alternative topologies are sought, and the one with the highest merit is the one passed on to the geometric finishing stage.

5 Future Work

It is planned to attempt to extend the range of sketches which can be interpreted by allowing non-trihedral vertices. These currently fail line-labelling, being identified and rejected in those cases where the non-trihedral vertex is fully visible.

It is also planned to attempt to extend the range of permitted input to allow sketches of objects with through holes and hole loops, and to extend the scope of reconstruction by partitioning difficult sketches into smaller, categorisable subsketches.

We are considering methods for inferring the presence of non-trihedral vertices at the sketch boundary or in the hidden part of the object if this makes the most plausible interpretation of the sketch.

Further in the future, a system which can deal with simple curved objects may be attempted.

6 Acknowledgements

The authors are grateful to Unigraphics Solutions Inc. for providing Parasolid to the authors for use in this research.

References

[1] L.Bauer, *Elimination with Weighted Row Combinations for Solving Linear Equations and Least Squares Problems,* Handbook for Automatic Computation **II**, Linear Algebra, eds. J.H.Wilkinson and C.Reinsch, Springer-Verlag 1971

[2] G.Butlin and C.Stops, *CAD Data Repair,* 5th International Meshing Roundtable, Sandia National Laboratories, 7–12, 1996

[3] E.W.Dijkstra, *A Note on Two Problems in Connexion with Graphs,* Numerische Mathematik **I** 269–271, 1959.

[4] S.W.Draper, *Reasoning about depth in line-drawing interpretation,* PhD Thesis, Sussex University, 1980.

[5] A.Gibbons, *Algorithmic Graph Theory,* Cambridge University Press, 1985.

[6] I.J.Grimstead, *Interactive Sketch Input of Boundary Representation Solid Models,* PhD Thesis, Cardiff University, 1997.

[7] G.E.Hinton, *Relaxation and its Role in Vision,* PhD Thesis, Edinburgh University, 1978.

[8] D.L.Jenkins, *The Automatic Interpretation of Two-Dimensional Freehand Sketches,* PhD Thesis, Cardiff University, 1992.

[9] R.R.Martin and D.Dutta, *Tools for Asymmetry Rectification in Shape Design*, Journal of Systems Engineering **6** 98–112, 1996.

[10] A.Selinger and R.C.Nelson, *A Perceptual Grouping Hierarchy for Appearance-Based 3D Object Recognition*, Computer Vision and Image Understanding, **76**(1), 83–92, 1999.

[11] P.A.C.Varley and R.R.Martin, *A System for Constructing Boundary Representation Solid Models from a Two-Dimensional Sketch*, Proc. GMP 2000, IEEE Press, 2000.

[12] P.A.C.Varley and R.R.Martin, *Constructing Boundary Representation Solid Models from a Two-Dimensional Sketch — Sketch Categorisationand Frontal Geometry,* 1st Korea-UK Joint Workshop on Geometric Modeling and Computer Graphics, 2000.

[13] P.A.C.Varley and R.R.Martin, *Constructing Boundary Representation Solid Models from a Two-Dimensional Sketch — Geometric Finishing*, 1st Korea-UK Joint Workshop on Geometric Modeling and Computer Graphics, 2000.